

# The Hybrid Music System for the BBC Microcomputer



## USER GUIDE



# The Hybrid Music System for the BBC Microcomputer



## USER GUIDE

First published 1986

Copyright (C) 1986 Hybrid Technology Limited. All rights reserved.

Neither the whole nor any part of the information contained herein may be adapted or reproduced in any form without the prior written approval of Hybrid Technology Limited.

Hybrid Technology Limited  
Unit 3, Robert Davies Court  
Nuffield Road  
CAMBRIDGE  
CB4 1TP

Issue 1

# Contents

1 Introduction	5
Part 1 - Tutorial	7
2 Using AMPLE	9
3 Mixing pieces	15
4 Entering music on the staff	21
5 Editing words in text form	41
6 Programming with AMPLE notation	45
7 Making instruments	57
8 Building pieces	77
9 Further use of AMPLE	87
Part 2 - Reference	91
10 AMPLE commands	93
11 AMPLE music words	101
12 Special effects	125
13 Sound instructions	127
14 Instruments	137
15 Errors	143
16 Editor controls	149
17 Glossary	151
Index	158



# 1 Introduction

Welcome to the exciting world of computer music making! The Hybrid Music 5000 Synthesiser is your gateway to a fascinating adventure of discovery in music and sound.

The Music 5000 package turns your BBC Micro into a computerised music production studio - with the help of four different studio departments, you can take a piece of music all the way from a sheet score, an idea of your own, or one of the supplied examples, through arranging, playing and 'live' mix-down to a finished performance complete with realistic instruments and studio effects.

The Synthesiser unit is a powerful stereo digital synthesiser capable of a vast range of sounds that includes natural-sounding, synthesiser-type, and new abstract instruments - up to eight different ones playing together. You can make your own instruments by modifying any of the standard selection, or designing new ones from scratch, choosing from hundreds of waveforms, envelopes and sound control settings.

The Music 5000 is controlled through AMPLE (it stands for Advanced Music Production Language and Environment), the most powerful music language ever created for a microcomputer. Its heart is the AMPLE Nucleus ROM - this provides all the basic functions for programming and playing music, including a built-in computer music notation and interfaces to system extensions. On top of this is Studio 5000 - an advanced and easy-to-use environment of integrated editors, menus, and control modules in which you work to build music and design sounds.

This manual has been carefully written for the beginner to computer music. The Tutorial Section takes you through the process of entering, editing, arranging, mixing, and performing musical pieces, and selecting, modifying and designing the musical instruments that the players will use. For further information on any subject you turn to the Reference Section, where descriptions of all the commands, symbols and controls are laid out in an easy-to-access form.

But it doesn't end there. AMPLE is also a full-spec programming language of enormous flexibility - to the standard data and control structures are added advanced music programming features such as concurrency and random access to the time domain. Those who are prepared to delve into this side of the system are rewarded with the power to create computer-composed music, custom music notations, new special effects, interactive musical machines ... all the way up to user interfaces and new languages. In fact, Nucleus and Studio 5000 are themselves almost entirely written in AMPLE! Though all these

## Introduction

programming facilities are present in the standard system, they are beyond the scope of this Guide - they are catered for by a separate manual: the AMPLE Programmer's Guide.

And finally, we come to expansion. From the ground up, AMPLE is designed for expansion - the Nucleus ROM already incorporates system interfaces that allow further hardware units to be added in a completely integrated fashion, complete with matching Studio 5000-style software. If you have returned your User Registration slip to us, we will send you full information of these and other future developments as soon as it is available.



## **Tutorial section**



## 2 Using AMPLE

### before you begin

You should have a system set up as described in the Installation Guide, including a system disc or cassette created from the issue disc or cassette supplied. Turn on your computer system, Music 5000 and amplifier, and continue.

### starting the system

To start the system from disc, place the system disc in the top drive (drive 0) and press and release BREAK while holding down SHIFT.

To start the system from cassette, load the cassette into the recorder at the start of side 1 and enter:

```
CHAIN""
```

Set the recorder to 'play', and follow any further screen directions.

After a moment, you will see the AMPLE Nucleus sign-on message on the screen, and the system will continue to load. When this is complete, the Studio 5000 Main Menu will appear.

If the menu fails to appear, refer to the Installation Guide supplied with the package.

### playing pieces

A variety of example pieces are supplied, ready for you to play. To see the selection, press f9 (the right-most red key - take care not to press its near-neighbour BREAK). The 'Jukebox' will load and display a menu of pieces. This is the first of three menus - to page through them all, you simply press the RETURN key.

To play a piece from a menu, press the down key (one of the arrow keys on the right of the keyboard) to move the green stripe (the menu cursor) down to the piece you want. If you go past, press up to get back. When you've positioned the cursor, press RETURN - the piece will automatically load, present its title display, and begin to play.

You can return to the Jukebox menu at any time by pressing f9 again.

To return to the Main Menu from a Jukebox menu or from a piece, press f0, marked 'MAIN' on the '%' line of the function key strip.

## using the keyboard

When using AMPLE, the computer keyboard works much as you would expect, with a few exceptions.

Most of the time you work in an Editor, where the keys have their own special functions (described later on, and summarised in the chapter 'Editor controls'). Editors respond to lower-case letters, so you should leave the CAPS LOCK light off.

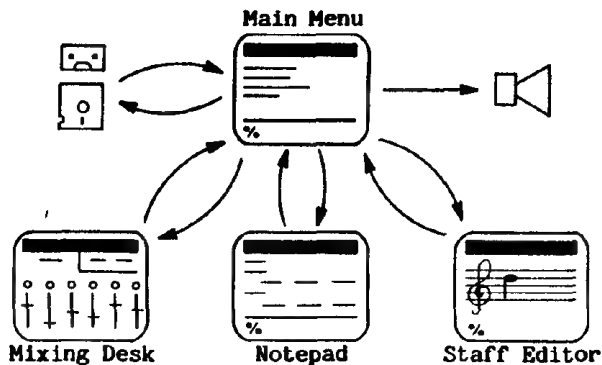
When typing commands, you use the SHIFT key as normal to enter upper-case letters. You use DELETE to remove mistakes and RETURN to end each command line.

You should never need to press BREAK, and it is a bad idea to do so. If your computer has a BREAK lock, make sure it is engaged. If you should press BREAK by mistake when working on a program, you should save it straight-away and then restart the system.

The function keys provide the commonly-used commands shown on the key strip on the '%' (bottom) line. You can use the corresponding key instead of the command wherever you need it at the end of a line.

## using the menu

The Main Menu is the Control Centre of the system. Through it, you access the Studio 5000 Editors, the 'departments' of the studio, and key functions like loading, saving and playing pieces.



To select an option, you use the up and down keys to position the menu cursor, and then press RETURN. Some options return to the menu, while others exit it entirely. To call the Main Menu back, just press f0.

Here's a run-down of the options on the Main Menu (they are fully described later on):

Run program	run the piece (AMPLE program) currently in memory
Notepad	enter the text and panel editor; called 'Notepad'
Staff Editor	enter the Staff Editor, for editing music on the staff
Mixing Desk	enter the Mixing Desk editor, for mixing-down pieces
Ready system	get the system ready for operations, stopping any music
Load program	load a program from a file, replacing the existing one
Save program	save the current program to a file, for later reloading
Catalogue	display the catalogue of the disc or cassette
New program	discard the current program, for entry of a new one
Commands	switch to command mode, for additional system operations

## commands

In addition to the Main Menu, AMPLE has a large set of direct commands. These include the menu options (the simpler ones actually show their commands when selected) and further command-only functions. The chapter 'AMPLE commands' lists the most important ones.

Commands are entered at the '%' prompt, which appears in the **command area** of the Main Menu. You switch to the command area from the Main Menu by either selecting 'Commands', or pressing TAB. To get back to menu mode, hit TAB again. This convention applies throughout the system - TAB always goes to command mode and back again.

You always press RETURN at the end of a command line to enter it. Except in special cases, you can put more than one command on a line, but put spaces between them to avoid confusion. You must type commands in upper or lower case as specified. You can abbreviate most of them with a dot, but try the abbreviation before relying on it.

Any command line starting with a '\*' is sent to the operating system, so you enter '\*' commands in the usual way.

You will inevitably come across error messages - announcements starting with '!' that appear when the system cannot carry out your requests. Often, the problem will be obvious - when not, look the message up in the chapter 'Errors' for an explanation.

## programs and words

As in other computer languages, the complete sequence of AMPLE instructions to do a particular job is called a **program**. In AMPLE, this job is usually playing a piece of music.

There is one program in memory at all times, even if it is a clear, empty program. The program can be saved to disc or cassette using the Menu option (or the equivalent command) and reloaded in a similar way later on. When you load a program, it replaces the one in memory.

The basic unit of an AMPLE program is the **word**. Each word has a small job of its own (you can look on it as a sub-program) and is created, edited, and if necessary deleted independently of others. AMPLE commands and instructions are also words, so to make a distinction, we call words in the program **user words**. System word names are in upper case, and user word names are largely lower case, so there is rarely any confusion.

The complete sequence of instructions in a word is called its **definition**. They can be anything from musical notes to sound settings. When you build a piece of music, all the information you enter through the various Editors - musical parts, instrument designs, mix controls etc. - is stored as user words, and therefore becomes part of the program.

You can always view the word contents (or **definition**) in the form of computer program text, even if it is a staff score, for example. This means that you can make, print and type-in complete piece listings, directly at the keyboard (see the command WRITE).

AMPLE words are more than just storage devices, because each one can be used as an instruction in a new word, just like an AMPLE system word. For example, you could create common musical sections as individual words, and then arrange them in sequence in a further word to make the complete part. In this respect, words are like procedures in other languages, but they are much more versatile: a word could be anything from a new musical symbol to a command to play the complete piece.

## concurrency

One of the most important features of a computer music system is 'concurrency' - the ability to do more than one thing at the same time. This lets you play musical parts in parallel and, most importantly, control the music from the keyboard at the same time. AMPLE is fully concurrent, offering up to 10 parallel players (musical parts) plus the master player that runs the keyboard and screen.

In an ordinary computer language such as BASIC, when you enter RUN to run the program, the system goes away and does not accept commands until it is finished. In AMPLE, RUN starts the program running and then returns control immediately; the system continues to accept commands while the program runs. This means that:

- \* the music plays 'in the background' while you continue to work
- \* you can control and visually monitor the music as it plays
- \* you have a menu option and command to stop the program
- \* an error in the music may interrupt you at the keyboard

The last happens only if there is a fatal fault in the music, but this could occur right in the middle of you typing a command.

## editors

Editors are AMPLE's 'user interfaces'. Each one is a self-contained application program designed for a particular stage of the music creation process.

When you select an Editor from the menu, it takes control to provide a command mode with extra commands specific to the editor, and its own **edit mode**. You switch between these with TAB. In edit mode, you enter and edit data, for example music on the staff. For some jobs, you can edit 'live' (while the music is playing). You use command mode to store and recall the data as words, rather like files stored and recalled from BASIC. The words are then be combined at a later stage to make a complete piece.

The three editors are:

Notepad	for instruments and textual words
Staff Editor	for music on the staff
Mixing Desk	for 'mixes' - settings on musical voices

You can return to the menu from any Editor's command mode by pressing F0.

## modules

Studio 5000 exists as a set of **modules** - integrated system programs which are loaded into memory either permanently when the system starts, or, as in the case of Editors, just when required. Modules are held under directory 'M' on the system disc.

Editor module loading is carried out automatically by the Main Menu. On disc systems, the modules must be on the current drive or the drive specified for modules (see the chapter 'Further use of AMPLE'). On

## Using AMPLE

cassette systems, you must follow loading instructions when they appear.

### **use with cassette**

To make module loading quicker, you can note the tape counter positions for the three Editor modules, and wind to these points to load them.

The Main Menu keeps each Editor module until another editor is selected, but discards it to make room on each of the 'Run program', 'Load program' and 'New program' options. You can avoid it being discarded by using the direct command versions of these options (displayed on selection), but this may leave insufficient memory for large programs.



## 3 Mixing pieces

'Mixing' is the creation of the final sound of a piece by combining each of the musical voices in the score with instruments and sound settings, and setting overall parameters such as tempo and tuning. You can start using the system straight-away by mixing selected example pieces.

The Mixing Desk is one of the most exciting 'departments' in the Studio. It is a simulation of a real studio mixing desk with volume faders, stereo pan knobs, visual monitors etc., all of which work on the music while it plays, in 'real time'. Being part of a completely computerised studio, it also gives you control over parameters impossible on a real mixing desk, including instrument selection, tempo, overall tuning and tape-recorder style pause and fast wind.

### trying out the Mixing Desk

Before calling-up the Mixing Desk, we'll listen to an example piece to try it out with. Select 'load program' from the Main Menu, and enter:

manor (RETURN)

Select the 'Run program' option to start the music playing.

To call-up the Mixing Desk, select it from the menu. After a moment, the Mixing Desk screen will appear with a blank command area at the bottom. To display the full screen, press TAB.

Most of the screen is taken up with eight identical columns of controls, one for each of the eight voices. At the top of the screen, there are two groups of overall controls, one of them contained inside a box on the right. This display may look daunting at first, but you will soon understand how it works.

If you press TAB again, the command window re-appears. TAB switches between **command mode**, where you enter commands at the '%' prompt, and **edit mode**, where you adjust mixing desk controls on the full screen. You can return to the Main Menu by pressing F0 in command mode.

## playing a piece on the mixing desk

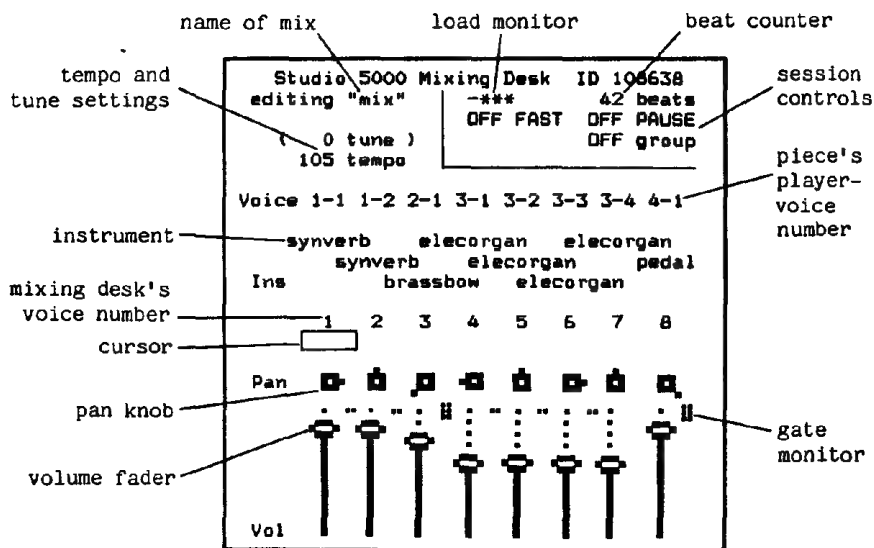
To start the example piece playing, make sure you are in edit mode (the full graphics screen) by pressing **TAB** if necessary, and press the **r** key. The title appears, the music starts, and the controls and settings change to the mix of the piece.

Notice the voice numbers 1 to 8, each with the name of its instrument lined up above. As the music plays, you'll see:

- \* at top right, the beat counter counting up like a tape counter
- \* at top centre, a bar moving left to right, monitoring the load
- \* two thirds down, eight gate monitors blinking along with the music on the eight voices

## the mixing desk screen

The items that show on the Mixing Desk while 'manor' plays are as follows:



We can easily experiment with the controls when the music has finished.

First let's try the cursor (a solid magenta bar): you use the normal cursor keys to move this to the control you want to change. You can put it on an instrument, below a voice number, or on one of the controls in the top left of the screen. With the cursor below a voice

number, you can play a note on it by pressing RETURN - try it now on each of the voices.

To adjust a control, you use the cursor keys in combination with the SHIFT key. To move the volume fader up or down, press SHIFT up or SHIFT down, and press RETURN to hear the new volume. To move the stereo position, use SHIFT left and SHIFT right - the pan knob above the fader will rotate accordingly.

## a 'live' mix

Now try adjusting levels 'live' on the music - press r to start, and move through the voices with left and right, setting the volumes and pans to your taste. You might like to make the tune on voice 3 a bit quieter, or move all the organ voices (4 to 7) off to one side.

You can easily change instruments while the music plays, rather like sneaking up to a player and quickly switching his instrument in the middle of playing. To change an instrument, move the cursor to it and press SHIFT. The instrument menu appears with the names of all available instruments and a flashing block cursor on the instrument currently in use on this voice. With SHIFT still down, use the normal cursor keys to move the block cursor to a different instrument, and then release it to select that instrument.

You could change the 'brassbow' that plays the tune on voice 3 to 'Wha', a human voice sound, or replace 'elecorgan' on voices 1 - 4 with 'Organ', a grand church organ. You may want to re-balance the volumes to suit the new instruments.

You could either work through the alternative instruments on the playing music, or wait until it finishes and try them out 'in solo' with the RETURN key. Those instruments with upper-case initials are preset instruments (built-in to the system) whereas those with all lower-case letters are part of the program for this piece.

## keeping the mix

The piece always starts with the same mix, even if you changed some of the settings the previous time it played, because the mix is stored as a word in the program. To replace this mix with one of your own, you use the command MAKE.

When you've settled on a mix that you want to keep, press TAB to switch to command mode, and enter

MAKE (RETURN)

## Mixing pieces

Some AMPLE mix instructions appear on the screen as the settings of the desk are stored as a word, replacing the previous definition. Now when you press r to run the music (dont forget to go back to edit mode), it will play with your mix.

To save your re-mixed version of 'Manor' return to the Main Menu by pressing f0 in command mode, and use the 'Save program' option. Choose a different name from 'manor' to avoid replacing the original.

To return to the old version, reload the piece using the 'Load program' option on the Main Menu.

## tempo and tune

Since in our studio the players are computerised as well as the Mixing Desk, we can change the overall tempo and tuning as easily as volume and pan. Tempo and tune are at the top of the screen - you can reach them with the cursor by moving up from any voice.

The tempo number is in crotchets per minute. To change it by single units, you use the SHIFT up and SHIFT down keys, and to change it by tens you use SHIFT left and right.

The tune number is in sixteenth-semitone units, measured from the normal tuning at zero, so for example -192 is down by one octave. To adjust it, you use the SHIFT cursor keys just like tempo.

After adjusting either tempo or tune, you can instantly return it to the value it had when you moved to it by pressing COPY.

MAKE stores the tempo and tune settings along with the voice controls, so they are re-called automatically when the piece is played.

## pause and fast

The box in the top right corner of the screen contains session controls - ones that help in mixing and are deliberately not stored as part of the mix. The first two are 'pause' and 'fast'.

**Pause** and **fast** are rather like tape recorder controls: 'pause' stops the music temporarily and 'fast' fast winds through it.

The space bar controls 'pause' - pressing it once puts the music on pause, shown by the ON PAUSE indication in the box. Pressing it again takes the pause off. While the music is paused, you can adjust controls and use the RETURN key as normal - this makes pause useful for detailed examinations of instants in the music. Also in the paused condition, pressing '/' will step the music on a beat at a

time.

The f key controls 'fast' - when you press this, the music runs at top speed until you release it. 'Pause' takes precedence over 'fast'.

## groups

The other session control is **group**. This groups all voices on the same player together so that they are adjusted as one - if you adjust a setting on one, they all change. Pressing g toggles between 'ON group' and 'OFF group' (the normal state). You can demonstrate this on the organ voices of 'Manor', using it to make them all louder or softer together. The voices are not permanently grouped - you could set the volumes together as described, and then return to 'OFF GROUP' to spread the voices across the stereo field.

When group is ON, RETURN plays the whole group of voices. You can still play individual voices by using the number keys 1 to 8: they work just like RETURN but on their respective voices only.

You can tell which voices will be grouped by looking at the 'player-voice' number for a given Mixing Desk voice - this is the two-digit number above the instrument names. The first digit is the player number, and the second digit the number of the voice in that player. A group includes all voices on the same player.

## multiple mixes

The Mixing Desk lets you keep different mixes in the same program and recall them while the music is running.

To keep a new mix in addition to the main one, we make it as a separate word under a different name. At the top left of the screen, you'll find the message 'editing "mix"' - 'mix' is the name of the main mix. To set the name to something else, you use the NAME command, entered in command mode, with the new name enclosed in double quotes before it, for example:

"mix2" NAME

% (the space is optional)

You now enter MAKE as before to create mix2. When you go back to edit mode, you will see 'editing "mix2"' at the top.

To call up your alternative mix, you simply enter its name as a command:

mix2

## Mixing pieces

To switch back to the original, you enter 'mix' in the same way.

Try creating a few different mixes and switching between them. The names should be 'mix' with a single digit on the end. Note that the piece always starts with the mix called just 'mix'.

In a later chapter, we will see how to call-up mixes automatically on cues in the music.

## the dictionary of words

When you made 'mix2' as a word, it was added to the list of user words. This is called the user **dictionary**. You can display this list with the command

```
SHOW
```

You'll see the names of all the words in the program, arranged in alphabetical order. Some of them are musical sections, some are instruments, and in the middle, under 'm', you should find your mixes.

You can remove any mix word that you don't need any more with the DELETE command (don't confuse this with the DELETE key). DELETE takes the word name in the same way as NAME, for example:

```
"mix3"DELETE
```

You can check the dictionary with SHOW to check the word has gone.

## trying other pieces

'Manor' is one of the simplest example pieces as far as mixing is concerned - it has just one mix which stays fixed throughout the piece. Some of the others do all sorts of things while they play - switch instruments, move sounds, bring them in and out: all these changes are stored as separate mixes, called up on cue by the program. Each time a new mix appears, its name pops-up in the 'editing' message and the controls automatically switch to their new settings, even if you were in the middle of moving one of them at the time!

Spend some time on the Mixing Desk, playing other example pieces, and creating your own re-mixed versions. You may see items that you don't recognise, like controls in brackets: just ignore these for now. With any luck, you'll soon feel familiar with the Desk and be ready to mix your own pieces when the time comes.

That's it on the Mixing Desk for the moment. We will come back to it in the chapter 'Building pieces'.

## 4 Entering music on the staff

Now we come to the musical information itself - the 'score' part of a complete performance. In this chapter we will look at the most familiar form of score available in AMPLE - staff notation.

Using the Staff Editor, you enter single-part tunes or chords by typing the notes and other symbols of conventional notation on to a staff. You can then have the music play directly from the staff, or make it into an AMPLE word from where it can be played on command.

To create a multi-part piece, you enter each section of the music as a separate word and later, using another editor, assemble these section words to build the complete piece. Parts, repeats, alternatives, introductions etc. are all catered for at the building stage, leaving the staff free for the details of the notes themselves.

### calling-up the Staff Editor

Before starting this session, save the existing program if you want to keep it, and select 'New program' from the menu (or enter the command NEW) to start afresh. This makes sure the words you make don't get muddled up with those of a left-over program. (In the future when you are adding words to an existing program to build a complete piece, you will call-up the Staff Editor without clearing the program.)

To start this session, select 'Ready system' from the menu (or enter the command READY). This stops any program that is playing, frees all the voices and additional memory in use, and makes some important initial settings.

Select 'Staff Editor' from the main menu. After a moment, the Staff Editor screen appears, leaving the % prompt near the bottom of the screen. You are now in the Staff Editor's command mode, from where you can type commands as normal.

To enter and edit music, you press TAB to enter edit mode - the command mode cursor disappears and an edit mode cursor (a solid block) appears above the staff. Pressing TAB again returns you to command mode.

Remember that TAB switches you between command mode, in which you enter commands at the % prompt in a window at the bottom of the screen, and edit mode, in which you work on the staff.

Entering music on the staff

To return to the main menu, make sure you are in command mode and press f0.

## entering notes

Make sure you are in edit mode, and that the CAPS LOCK light is off.

The screen shows the large double staff on which you put symbols, with space above it for additional musical instructions. The '[' and ']' characters mark the start and end of the music on the staff - at the moment, there is none. The cursor indicates the item you are working on, and the point at which new symbols are inserted.

RETURN enters a note, so try pressing it now. The cursor moves to the right, ready for another note.

You can alter this note by first moving the cursor to it (directly above it). Do this by pressing the left key. You can now change its pitch with the SHIFT up and SHIFT down keys - hold SHIFT and press up or down to make the note move up or down the staff.

If the cursor changes from the normal block to a line, you have pressed unshifted up by mistake. We'll see what this is for later - press unshifted down to restore the block cursor.

To change the length of the note, use the SHIFT left and SHIFT right keys - these change to the next shorter or longer of the standard note lengths, demisemiquaver to semibreve. (There is also a special length shown as a stem-less note - we will look at this later).

## entering a tune

If you press RETURN on this changed note, an identical note is inserted before it, moving the original note to the right. This is how you enter a string of notes. Think of the note you are on as a musical paintbrush - you set it to the pitch and length for the note you want, then press RETURN to 'paint' that note onto the staff and move the 'brush' on to the next position.

The DELETE key removes the note before the cursor - if you 'paint on' the wrong note, use this to remove it, and try again.



Now we will try an example tune. Before starting, make sure you have a clear staff by switching to command mode and entering

CLEAR

You should do this each time you start on a new tune in the Staff Editor.

Here's the example:



Notice the clef sign at the start and the vertical line at the end - we enter both these on the editor staff. The 'c' key inserts the clef, and the 'e' key inserts a thick end line (note: this is not a bar line).

Start by pressing 'c' for the clef. Both treble and bass clefs appear - the notes of this example go on the treble (upper) staff. Press RETURN to get a brush note, and then enter the notes of the tune. After pressing RETURN to 'paint on' the last note, press 'e' to insert the end line.

You'll be left with the unwanted brush note after the end line. The COPY key removes the symbol at the cursor, so press it now to remove the extra note.

You may have been tempted to miss out the last RETURN and leave the brush note as the last note, making the mistake of thinking of RETURN as 'going on to the next note', rather than painting on a note. If you did this, you'd have to move the cursor to get the end line in the right place. If you do it the way we recommend, you can enter complicated sequences easily with no cursor moves.

If you spot a wrong note after you've finished, use the left and right keys to move the cursor to it, and then adjust the pitch or length using the normal keys. You can remove an extra note with DELETE or COPY. To add a missing note, go to the note after the place where it should have been, press RETURN, then move back one and adjust the new note's pitch and duration.

As you move the cursor, the item number at the top left of the screen changes to show at what point in the tune you are.

Entering music on the staff

## playing a tune

To hear the tune, you first set up a voice with an instrument. This you do by going to command mode and entering

```
1 VOICES Upright
```

for example.

To play the tune, return to edit mode and press f1 play. The cursor disappears and returns when the Editor is ready for your next command. If you want to stop the music before it has finished, press ESCAPE.

You can easily play the tune on other instruments, for example

```
1 VOICES Vibglock      or      1 VOICES Panflute
```

To display a list of all available instruments, enter the command

```
"CHANS" FIND
```

To change the playing tempo, you use the =T instruction to set it in beats per minute - the normal setting is 125 crotchets per minute. To go twice as fast (250 crotchets per minute), enter

```
48, 250=T
```

The 48, indicates that the beat is a crotchet. To return to normal, enter

```
48, 125=T
```

## making into a word

You can store the music that's on the staff as an AMPLE word. This lets you:

- \* play it by entering the word name as a command
- \* include it as a section in a complete piece
- \* save it on disc or tape
- \* edit it in AMPLE music notation

The word is described as a 'score word', since it contains score information, distinguishing it from mix, instrument and other words.

Before making the word containing your tune, you give it a name with the NAME command. The name of the word you are making is shown at the top of the screen - since you started from scratch with CLEAR, it is currently 'newword'. To give a new name, go to command mode and enter

"ode"NAME

(You could choose just about any name of up to 15 lower-case letters).  
Now, to make the word, enter the command

MAKE

You will see AMPLE music notation displayed as the word is made, after which, the % prompt returns. If you enter SHOW, you will find 'ode' has been added to the list of user words.

You play the music from the word by entering

ode

and try different instruments by entering

1 VOICES Upright  
ode

and

1 VOICES Panflute  
ode

(Your music word should be entered as the first item on a new line, so that it starts properly.)

You can save the word 'ode' (with any other user words that are present) as a program, either with the 'Save program' option on the main menu, or by entering the SAVE command directly:

"odeprog"SAVE

To reload the program, use 'Load program' from the menu, or the direct LOAD command:

"odeprog"LOAD

## editing an existing word

When you enter CLEAR to start another tune in the Staff Editor, the music on the staff is lost. If you made this music into a word, you can get it back onto the staff for editing.

## Entering music on the staff

To edit an existing word in the Staff Editor, use the command GET, for example

```
"ode"GET
```

After a moment, the Staff Editor display appears with the music of the named word, leaving you in command mode.

When you get a word, the Editor automatically sets the 'making' name, so that you can carry out changes and remake the same word without having to set the name. To make another word containing the altered (or the same) music, first use NAME as before to change the name.

## rests

A rest is inserted by the '^' symbol (top right of the keyboard), just like a note is inserted by RETURN. When you are entering a sequence and come to a rest, just make the brush note the right length, and enter '^'.

If you need to change the length of an existing rest, or insert a new one in an existing sequence, you adjust its length using the SHIFT left and SHIFT right keys as normal.

## time signatures and bar lines

In the Staff Editor, bar lines make the music easier to follow and point out any bars that don't add up to the correct length. They have no effect on the sound. When entering from sheet music, you normally include the bar lines, but you can leave them out if you want.

You insert a bar line by entering the '|' symbol (top right of keyboard). Each time you come to a bar line in the sheet music, you press '|' and the line goes before the brush note. Don't forget the last bar line - it goes just before the thick end line, so together they look like the final (double) bar line found on the written music.

Bar lengths are not checked unless you add a time signature. This usually goes between the clef and the first note or rest, after any key signature that is present. Three pre-defined time signatures are supplied:

signature	key
6/8	d (for duple)
3/4	t (for triple)
4/4	q (for quadruple)

To insert one of these, just press its key.

You can adjust both the top and bottom numbers to get any of the other possible signatures. Move the cursor to the time signature and use the following keys:

<u>SHIFT right</u>	increase bottom number
<u>SHIFT left</u>	decrease bottom number
<u>SHIFT up</u>	increase top number
<u>SHIFT down</u>	decrease top number

Once you have a time signature, any bar whose length doesn't add up to the correct total will be spotted at the bar line when it plays - the error message 'Bad bar' appears, leaving you in command mode. (Don't be confused if the music has not yet reached this point - the system checks a few bars ahead.) When you return to edit mode, the cursor is positioned at the bar line that found the fault.

When you come to entering sections of a complete piece, you should play each section from the staff before making into a word, just to check bars.

Many tunes start and end in the middle of the bar. They are written out in sheet music with the first and last bars deliberately short of the full length. When entering these, you normally make each incomplete bar up to the full length with rests. The Staff Editor allows you to put a new time signature at the start of any bar, so you could alternatively give each short bar its own time signature setting, not forgetting to insert the normal one for the next bar.

## accidentals

The following keys insert accidental signs:

key	accidental
+	sharp
-	flat
=	natural

When entering music into the Staff Editor, you always type the symbols in the order they appear on the printed staff, and they will appear in the same order on the screen. This means that when you come to a note with, for example, a sharp before it, you enter the sharp (+) and then the note. To enter a double sharp or flat, you press the key twice.

Most printed music uses a shorthand whereby any accidental you see actually also applies to all further notes of the same letter name (including those in other octaves), up to the next bar line. In the Staff Editor, each accidental applies to the next note only, so you must add the accidental before each note that requires it.

Because of this shorthand, you will sometimes find a natural sign used to cancel the sustained effect of a sharp or flat. You don't need to put this natural sign in on the Staff Editor, though it does no harm.

## key signatures

You can have any of the full range of key signatures: up to seven sharps and seven flats. You normally enter the key signature after the clef, and before the time signature and first note or rest. After pressing 'c' for the clef, press '+' once for each sharp you want, or '-' once for each flat you want, and then press 'k'. 'k' converts the line of accidentals into a key signature with sharp or flat signs correctly positioned on both staves. For example, to enter a key signature with two sharps (D major or B minor), enter:

++k

If you make a mistake, use DELETE to remove the whole key signature, and try again.

You may find that a non-standard key signature (one with an odd sequence of sharps and/or flats) is more convenient for some pieces, for example, one that includes the sharpened seventh of a minor key. You can make any non-standard signature by putting in a seven-sharp or seven-flat signature (or both, one after the other), and then deleting the signs you don't need. Don't leave any spaces between signs in the key signature.

You can enter a new key signature at any point in the music, and there is no specific limit on the number of key signatures.

## spaces

You can insert spaces in the music to match the layout of long and short notes in the printed music, or just to make the tune easier to follow. It's a good idea to put spaces between each group of notes, particularly if you aren't using bar lines.

To insert a space, simply press the space bar. You can use this like a typewriter space bar when entering music, or add spaces once the music is entered.

## moving about

If by now you have entered a tune which is more than one screenful, you will have found that the screen moves along the staff when you get to the left or right edge. To move around a long tune more swiftly, you use the CTRL cursor keys:

<u>CTRL left</u>	move left by one screenful
<u>CTRL right</u>	move right by one screenful
<u>CTRL up</u>	move to start
<u>CTRL down</u>	move to end

You can find out how much room you have left on the staff by pressing CTRL down to move to the end, and reading the item number. There is room for 240 items.

## block editing

Many tunes have small groups of notes that are repeated, either exactly or with a few changes. To make these easier to enter, the Staff Editor has a block copy function. Before we look at this, remember that you should enter large sections separately so they can be called up for repeats at the later piece-building stage. If you find a repeated bit of more than a few bars, it's usually best to treat it as a section and make it into a word of its own, not make a copy on the staff.

To copy a group of notes, we use the f2 key, 'copy'. This makes a copy of the music from the cursor up to the end, and adds it on to the end. When you come to a small group of notes that repeats immediately, enter the notes once, move the cursor back to the start of the group, and press f2 copy to add a copy. Pressing it again copies the two to make four, and so on.

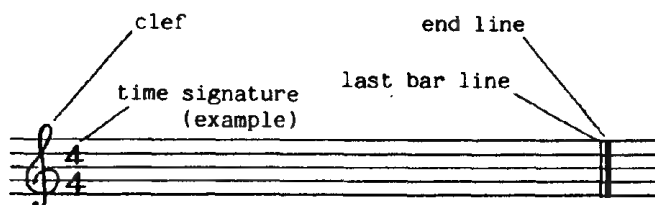
## Entering music on the staff

Often you find one group of notes, followed by another, followed by a repeat of the first - an 'ABA' structure. You can enter this by inputting A and B, copying both to make ABAB and then deleting the second B. Alternatively, use some forward planning: enter A, copy it to make AA, and then add the B in the middle.

To delete chunks of music swiftly, use the Editor's 'delete bar' function: pressing SHIFT COPY deletes everything up to the next bar line, or to the end of text if there is no further bar line.

## complete tunes

Here's a summary of the parts of a complete tune:



The clef initialises all music settings, for example, setting no key signature. It makes sure that the tune is not affected by what played previously.

The time signature indicates the length of the bar. It is often followed by a key signature.

The bar lines automatically check that each bar has the correct length. Don't forget the last bar line.

The thick end line finishes the last note, which would otherwise play on, waiting for a following note or rest. It goes straight after the last bar line, with which it makes a final double bar line.

## ties

As in written staff notation, the Staff Editor uses a curved line between notes (of the same pitch) to show a tie. To enter a tie, type the '~' symbol (top right of keyboard). When entering two notes tied together, just enter the first one, then press '~' for the tie, and then enter the second one. The tie can go anywhere between the two notes, but it is best to put it after any bar line and before any accidentals.

Where you want three or more notes tied together as one, just put ties



between each note and the next, as is shown on the written staff.

Remember that in the Staff Editor, accidentals apply to the following note only, so where you find an accidental before the first of a pair of tied notes, you must repeat it on the second of the pair.

## dotted notes and rests

When you come to a dotted note in entering music, after pressing RETURN to enter the note itself, simply press '.' to enter the dot. To add a dot to existing music, go to the symbol after the note where you want the dot, and enter '.' (like all other symbols, the dot is inserted before the cursor position).

Entering a dotted rest is just as simple: you enter the rest as normal with '^', then press '.' to add the dot.

To enter a note or rest with two dots, just add two dots.

You can add a single dot to any length note or rest from semiquaver upwards, and two dots to any length from quaver upwards.

## triplets and duplets

On the written staff, a triplet is shown with a curved line and a single figure 3 over or under the notes in the group. For clarity, the Staff Editor uses a separate figure 3 after each of the notes in the group, leaving out the curved line:



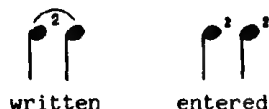
When inputting each note of a triplet, you enter it as written and then press '3' to add the figure 3. This makes the note play for two-thirds of its normal length, so three such notes add up to two normal ones.

To make an existing note into a triplet note, go to the symbol after the note, and press '3' to insert the figure 3. You treat rests in exactly the same way.

You may find more complicated triplets; with just two notes, with a dotted note, or with more than three notes. To enter these, follow the same rule - enter each note as written, and add a '3' after it. If the note has a dot, put the '3' after the dot.

## Entering music on the staff

Duplets work in a similar way. Where the written music has two notes with a curved line and a figure 2, you enter each note as normal but press '2' to enter a figure 2 after the note symbol itself.



You can put a '3' or '2' on any note or rest length, including the smallest (demisemiquaver) and all allowed dotted and double-dotted lengths.

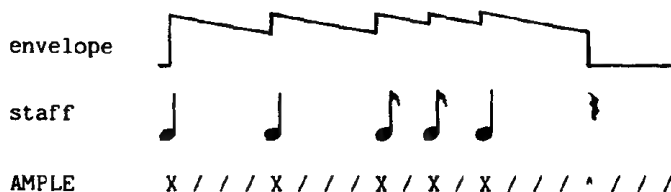
## an example

The following tune uses many of the symbols we have seen so far. (We have already added rests to make up the incomplete first and last bars.)



## note style


Normally the sound of each note is played for the whole of its written duration, so the notes play in a connected 'legato' fashion, represented as follows:




You can change the style of playing by shortening the time of the sounding portion, or 'gate period', of each note, so leaving a gap in the sound before the next one. (Note that with sounds that immediately die away to nothing, like drums, there is a gap anyway.)

To modify the gate period, we use the instruction 'Len' with a single number before it. 'Len' is an additional, preset instruction, as are preset instruments for example. As with all preset instructions, its name is in lower case with an upper-case initial.

A positive Len setting, for example, '12 Len', sets the actual gate period in time units, whatever the full length of the note. 12 units is the length of a semiquaver, so the effect of '12 Len' is like replacing each note with a semiquaver note plus rests to make up the full length:


envelope 


staff 

AMPLE X ^ / / X ^ / / X ^ X ^ X ^ / / ^ / / /

Positive Len numbers are used for short, staccato-like effects. Where you find a passage of staccato notes, choose a Len number that is half the length of the shortest note, as in the example above. At the end of the passage, use the instruction '0 Len' to return to normal, full-length notes.

A negative Len instruction sets the length of the gap between notes in time units, by reducing the gate period of notes of all lengths by a fixed amount. For example, -12 Len has the effect of a semiquaver rest before each note, taking its time from the previous note:

envelope 

staff 

AMPLE X / / ^ X / / ^ X ^ X ^ X / / / ^ / / /

Negative Len settings are very important for instruments which give no special emphasis at the start of notes - some organ sounds for example. Without a separating gap, you don't hear the start of a note unless the pitch changes: repeated notes of the same pitch play as if tied. For this sort of sound, you use a Len value of around -4 as the normal setting throughout the piece.

A small negative Len throughout can also suit instruments that do have an emphasised start to notes, such as piano-like ones. In piano-style music, you would temporarily switch back to '0 Len' for a group of notes marked with a slur.

Special musical instructions like 'Len' go above the staff, on the level of the cursor and start and end markers. They can be inserted at any point in the music, but since 'Len' is a musical effect, and therefore cancelled (set to 0) by the clef, there is no point in

## Entering music on the staff

putting it before the clef.

To enter an instruction above the staff, you first press the up key to go to 'text mode'. The cursor changes from the normal block to a line. Now when you type on the keyboard, the normal characters are inserted at the cursor. When you've finished, you press the down key to return to 'symbol mode'. The cursor changes back to the block and the keys return to their musical symbol functions. DELETE and COPY have the same effect in both modes.

To insert, for example, '12 Len', you press up to go into text mode, type it in, and then press down to get back to symbol mode. Remember to return to symbol mode, else you will find you cannot insert symbols on the staff.

## defining your own instructions

If your music needs frequent changes between note styles, it can be more convenient to define named equivalents to the 'Len' instructions, and use these instead. You make these as normal user words, with names of your choice, containing the instructions you want them to do. Simple definitions can be entered directly in command mode, for example:

```
"stac" [ 12 Len ]
"leg" [ 0 Len ]
"gap" [ -4 Len ]
```

The names are kept short for compactness on the staff. Once defined, you can enter their names as instructions above the staff precisely as you would 'Len'. The words are part of your program and so will be saved and loaded with the music that needs them.

You can define your own versions of any above-staff instructions, or sequences of them, in exactly the same way. To find out how to store them separately for use in future programs, see chapter 'Building pieces'. Two commands that are useful for user-defined instructions are FIND, to find all uses of a word, and TYPE, to display the contents, for example:

```
"Len" FIND    to find all uses of Len, including user instructions
               and score words that use it directly
"leg" FIND    to find all uses of the user instruction 'leg'
"leg" TYPE    to display the definition of 'leg'
```

**slurs**

As in written staff notation, the Staff Editor uses a curved line between notes to show a slur. This is the same symbol as the tie - if the note pitches are the same, it's a tie, and if they are different, it's a slur.

To enter a slur, type the '-' symbol (top right of keyboard). When entering two notes slurred together, just enter the first one, then press '-' for the slur, and then enter the second one. The slur can go anywhere between the two notes, but it is best to put it after any bar line and before any accidentals. Where you want three or more notes slurred together in a group, just put slurs between each note and the next.

The effect of the slur is to play the second note as a continuation of the first, without any gap and without restriking the envelopes.

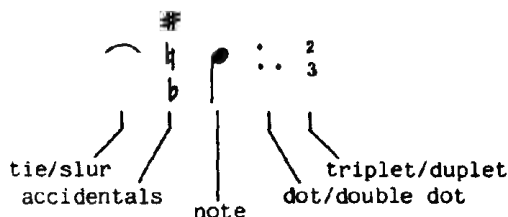
In piano parts, the slur symbol really means just 'legato' (no gap - a true slur cannot be played on a piano). In AMPLE, you produce this effect with a changed Len setting, not with a slur symbol.

On a flat organ-type envelope with a gap 'Len' setting, the effect of the slur is to remove the gap. If there is no gap on this type of unemphasised envelope, then the slur makes no difference.

On envelopes that quickly die away to nothing, you may not hear the second note at all - it doesn't make sense to slur a drum beat!

## summary of modifiers

Here is a diagrammatic summary of the various symbols that go before or after notes and rests, indicating the choice available in each position and the order in which the symbols go.

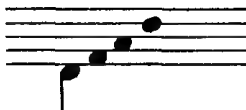


## chords

The Staff Editor shows chords in a simple 'exploded view' which makes it easier to enter and edit the notes individually. Each chord is represented as a normal main note followed by up to seven additional 'chord notes'. Each chord note is shown as a solid head with no stem. Here's an example:



written



entered

To insert a chord note you press '@', just like you press RETURN to insert a normal note. When entering a chord, you start at the bottom and put in the notes of the chord one by one, pressing RETURN for the first one, but '@' for the subsequent ones.

Try entering the example above, on the treble (upper) staff. Don't forget to CLEAR first, and include the clef at the start and the end line at the end.

## playing chords

To play chords from the staff, you set up a voice, with instrument, for each note in the biggest chord. For the example above, four voices are needed:

4 VOICES Vibglock

Note that you must include an instrument name - 4 VOICES alone is not sufficient.

## more about chords

The length of a chord is always that of the main note. Like a normal note, this can have dots and a figure 2 or 3. These extra symbols go right after the note, not after the whole chord.

You can add accidentals, ties and slurs to individual notes of a chord in the normal way. As in written music, ties work separately on each note, so you can hold over some notes while others change. To tie a complete chord, you put ties on all the notes. The same applies to slurs.

Rests take effect on all notes of the chord, as you'd expect. You are quite free to have a sequence with a mix of single notes and chords of different sizes.

You don't have to build chords upwards from the bottom note; you could make the top note the main note and move down, or put them in any order you like (remembering that the first note must always be the main, normal one). The main note always plays on voice 1 with the additional chord notes in order on voices 2,3,4, so you can choose which notes play on which voices, and then give each voice a different instrument.

Here's a short example of chords for you to try. It is from Tchaikovsky's 'Nutcracker Suite':

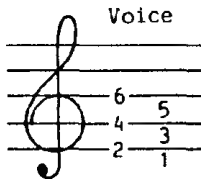


## percussion

The Staff Editor can be used to enter percussion scores as well as tunes and chords, for example a part to be played on a drum.

If the instrument has variable pitch, like the 'Yakbell' preset instrument, then you play pitches on it as normal. If the instrument has a fixed pitch, like 'Drum', then it doesn't matter what pitches you use because it always plays at its pre-defined, 'natural' pitch.

The Staff Editor lets you score for up to six percussion instruments on the same staff by using the lines and spaces for different instruments, rather than different pitches. You use the above-staff instruction 'Perc' to assign the six treble-staff lines and spaces from D to B, to voices 1 to 6:



You choose the instruments and put each one on a single voice, starting with voice 1 and going upwards. Each note on a voice's line or space will then play a hit on the corresponding instrument, using the instrument's pre-defined pitch.

The 'Perc' instruction must go after the clef because the clef sets the staff back to normal pitched playing.

Entering music on the staff

Where you want two or more hits on the same beat, you enter them in a chord. If this occurs often, it can be easier to write a dummy main note on every beat, and insert all the hits as additional chord notes. Put the dummy notes on the middle C line where they won't be heard.

## setting-up percussion voices

You usually make the final assignment of instruments to voices with the Mixing Desk, but for now you can do this directly by command. You use VOICES as before, but each voice is selected individually with VOICE to give it a different instrument, for example:

```
3 VOICES
1 VOICE Drum   2 VOICE Cymbal   3 VOICE Yakbell
```

You can store this as a word by defining it directly at the keyboard:

```
"kit" [
3 VOICES
1 VOICE Drum   2 VOICE Cymbal   3 VOICE Yakbell
]
```

Now, to set up the three voices for playing from the staff, just enter

```
kit
```

You can now play the percussion score by pressing f0 as normal.

## percussion patterns

Percussion parts often have a few fixed patterns, one or two bars long, strung together with many repeats to make a complete section. AMPLE lets you build sequences like this very easily - you create the patterns as separate words and call them up by name in the required order. For example, you could define 'pat1' as a main two-bar pattern and 'pat2' as a variation, and play them in sequence by entering

```
pat1 pat1 pat1 pat2 pat1 pat1 pat1 pat2
```

When you reach the piece building stage, you can use even more powerful facilities to build complete parts from elements like these.



## dynamics

To set the dynamic level of the music we use the AMPLE music notation instruction '=L', which in the Staff Editor is added above the staff.

=L has a number before it that determines the level in the range 0 (very quiet) to 64 (loudest).

To insert a level setting, press up, type in (for example) '50=L', and then press down.

The level is initially at maximum (64), so if you want the music to get louder later on, you must start with a quieter setting.

Here are suggested settings for the common staff dynamic levels:

pp	15=L
p	25=L
mp	35=L
mf	45=L
f	55=L
ff	64=L

You can usefully define your own named versions of these as described for 'len', but be careful in the choice of names - 'ff' would be normally interpreted in AMPLE music notation as two 'f'-letter notes, so use something like 'ff1' (for ff level) instead.

## accents

You can easily make a single note louder with the accent instruction. This is the single quote mark (centre top of the keyboard), which looks like this: ' .

The accent symbol is inserted above the staff, before the note to be accented. To enter a note with an accent, press

up ' down

and then input the note as normal.

The dynamic level starts off at the maximum so before you can hear the effect of an accent, you must reduce the level with an =L instruction. The normal accent adds 15 to the level of the note, so 49=L is a good general level.

To accent a whole chord, you put the accent symbol before the main note. You cannot accent individual notes of the chord.

## Entering music on the staff

The accent has no effect on tied or slurred notes - only a note that actually strikes can be accented.

You can change the strength of accents with the 'L instruction, inserted above the staff, usually right after the clef. For example, 20 'L increases the strength from the normal 15. Remember that 64 is the maximum total level.

## transposition

To enter pitches that are outside the range of the editor's double staff, you can use the AMPLE music notation word '@' above the staff. This is similar to the '8va' markings used for the same purpose in sheet music, but you give the number of semitones of transposition as a number before it. 12@ transposes up an octave, -12@ transposes down an octave, and 0@ returns to the normal pitch.

Where you find an 8va marking above the staff, insert 12@ before the first transposed note and 0@ after the last one. Similarly, use -12@ and 0@ where you find an 8va (or 8va basso) marking below the staff.

Where you find notes written outside the range of the Staff Editor's staff, you use the same method to enter them: place them an octave above or below the written pitch and use '@' to transpose accordingly.

## 5 Editing words in text form

So far we've looked at two editors - the Mixing Desk and Staff Editor - and seen how they are used to enter and edit their specific types of data (mix and score), storing the results as AMPLE user words. The third is 'Notepad', a multi-purpose editor that handles data in text form, that is, like a normal computer program or word processor document. Notepad is used for a variety of different types of data including instrument definitions, AMPLE music notation, musical section arrangements, and computations. As well as the standard text-editing facilities, it has some real-time control functions like those of the Mixing Desk and Staff Editor.

In fact, any AMPLE user word can be edited as text, even if created on the Mixing Desk or Staff Editor, because all AMPLE words have the same basic structure whatever their origin or purpose. This means that editors can freely exchange data, so you could, for example, enter music on the staff and then edit it in text form to add extra details.

This chapter introduces Notepad for general text-form word entry and editing, to prepare you for applications which use it for more specific tasks. These applications and the functions of Notepad that are specific to them are dealt with in following chapters.

### calling-up Notepad

Before starting this session, save the existing program if you want to keep it, and select 'New program' from the menu (or enter the command NEW) to start afresh. When you come to working on complete programs, you will usually call-up Notepad without clearing the program.

To call-up Notepad, select it from the main menu. After a moment, the Notepad screen appears, leaving the cursor at the % prompt in an area at the bottom of the screen. You are now in Notepad's command mode from where you can enter commands as normal.

To enter and edit text, you press TAB to enter **edit mode** - a cursor appears in the editing area (the blank central screen area). To return to command mode, you press TAB once more.

Remember that you enter commands at the % prompt in command mode, and enter text in edit mode - TAB switches you between the two modes.

You can return to the main menu at any time by pressing f0 in command mode, but this clears any text in the editor, so use it with care.

## entering and editing text

To enter text, you switch to edit mode and just type as normal - the characters go in at the cursor position. DELETE removes the last character you typed (the one before the cursor), and RETURN moves to the start of the next line.

You can move around the editing area using the four cursor keys - these let you move back to a mistake and correct it. Each new character replaces the one previously at that position ('overtypes' it). If you want to add or take away characters in the middle of a line, you can use character insert and delete:

<u>f8 insert char</u>	move the rest of the line right to make room
<u>f9 delete char</u>	remove the character at the cursor and close up

There are corresponding functions for lines:

<u>f6 insert line</u>	move the rest of the text down to make room
<u>f7 delete line</u>	remove the line, moving the rest of the text up

When you are inserting a character or a line, any characters pushed off the screen are lost, so take care.

Notepad has a copy function like the computer's own: using the cursor and COPY keys you can copy characters from another point on the screen, adding extra ones and deleting unwanted ones as you go. You press COPY to start copying and RETURN to finish. For example, to copy an existing line to the cursor position, you

- \* press COPY to start copying. A block cursor appears
- \* move the line cursor to the characters to be copied
- \* press COPY for each character. At this point, you can type in extra characters, or skip past any that you want to leave out
- \* press RETURN to end copying. The block cursor disappears

### Warning:

a fault in the BBC Model B's operating system allows you to move the line cursor out of the editing area. You should not attempt to do this. This fault is not present on other models.

To clear the text area ready for new text, you enter the command

CLEAR

## making a word

To make a word using Notepad, you use the command `MAKE` : this creates a word containing the text you have entered in the edit area.

If you try this now, you will probably get an error because `MAKE` first checks that the text is a valid word definition. So that you can try out the `MAKE` command without this happening, put a `%` sign at the start of each line - this marks the line as a comment, so you can put in any characters you like.

To make the word, switch to command mode and enter

```
MAKE
```

After a moment, the `%` prompt will return if the word was made successfully. If something goes wrong, for example there was a nonsense line without a `%` at the start, an error message appears and the word is not made. You can then correct the fault in edit mode, and try `MAKE` again.

## naming the word

When the word is made successfully, you can enter the command `SHOW` and you will see its name on the list of user words. Its name will be 'newword', the name that Notepad assumes until you specify one of your own. The current name is shown at the top of the screen in the message

```
got "" to make "newword"
```

You set your own name for the new word using the `NAME` command. `NAME` is preceded by the name you want in double quotes, for example:

```
"test"NAME
```

The name uses no upper-case letters, to avoid confusion with system words. It can be up to fifteen characters long, but you should try to keep it to nine or less to make various screen displays clearer.

After you've set the name to 'test', `MAKE` will make the word under the name 'test'. If you change the text and make 'test' again, the new one will replace the old. When you want two words with the same or similar contents, take care to change to a different name for the second one.

## editing an existing word

To alter the contents of an existing word, you first call it into the editor with the command GET. The name of the word goes first, in quotes, for example:

```
"test"GET
```

The contents of the word will appear in the edit area, and the top of the screen will change to show

```
got "test" to make "test"
```

indicating that the word you used GET on was 'test', and the word you are making is 'test'. You can now make your changes to the text and re-make the word without having to set the name with NAME.

If you use GET on a preset word - an instrument, for example - the 'to make' name will be set to 'newword' rather than the original name. This is because if you want a modified version of a preset word, you must create the new version as a user word - you cannot change the preset word itself. The 'got' part of the message reminds you which word you started with.

## 6 Programming with AMPLE notation

The AMPLE language has a built-in music notation of its own. Unlike staff notation which is graphic, AMPLE music notation is textual, that is, it uses the normal printing characters including letters and numbers, entered and displayed in normal computer form. It has equivalents of the usual staff graphic symbols, plus others which are not available on the staff but which are none-the-less important and useful. Any tune that can be entered on the staff can be simply and easily represented in AMPLE music notation.

You can use AMPLE music notation as an alternative to the standard staff notation. This will suit you particularly if:

- \* you do not read staff notation
- \* you have experience of a logical language such as a programming language, including BASIC
- \* you are interested in the mechanisms of music and not just the final result
- \* you want to produce large or complicated pieces

AMPLE music notation is also particularly suited to the types of music that go well on computers, including

- \* highly-structured music, having, for example, patterns and sections in sequences and variations etc.
- \* music in which the 'studio end' (instrument sounds, balance etc) is at least as important as the 'score end' (written elements).
- \* music which includes more than just the standard pitch, duration and loudness in the written score
- \* music which actually uses automatic decision-making processes in the performance, including computer-composed music

This includes much modern music; pop, rock and contemporary classical.

AMPLE notation has the following advantages over the staff notation:

- \* it is simpler and more logical in design, and easier to learn
- \* it has important additional musical instructions
- \* you can enter it directly from the keyboard, even as commands
- \* you can print it out as and enter it directly from a listing
- \* it is about ten times more compact on the screen
- \* it is more economical in memory, so you can have bigger pieces
- \* execution is faster, since it represents music more directly
- \* it is integrated with more-advanced programming functions

You enter and edit music in AMPLE notation using Notepad.

## calling-up Notepad

Before starting this session, save the existing program if you want to keep it, and select 'New program' from the menu (or enter the command NEW) to start afresh. When you get on to building complete programs, you will switch to Notepad to edit an existing word, so you won't clear the program first.

To start this session, select 'Ready system' from the menu (or enter the command READY). This stops any program that is playing, frees all voices and additional memory in use, and makes initial music settings.

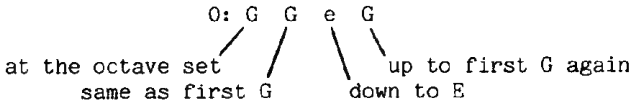
## basic music words

We'll go straight in and have a look at a tune in AMPLE music notation. The tune is 'Oranges and Lemons':





pitch. So, for the first four notes in the example:



If you want an interval of more than an octave, you put a ! before the second note. This jumps an extra octave in the appropriate direction.

## hold

a 'slash' (divide) character. Each hold makes the previous note one beat longer, so the 'e/' is a note twice as long as 'e'. 'e//' would be three times as long. You can get any note value you like with the right length setting and holds.

## space

just used to separate groups of characters for clarity. In this example, each group corresponds to a bar - six quaver beats. The spaces after the length and octave settings are not essential either.

## rest

'^' character (top right of keyboard). Rests are the same length as notes, and can also be extended by holds. This rest is included at the end to silence the last note, since notes (and rests) play on forever until the next one.

## trying out music

You can enter a line of AMPLE music notation directly as a command so that it plays as soon as you press RETURN, but it is often easier to use Notepad. If you enter the music into Notepad, you can make changes without re-typing, and play it at any time.

So that you will be clear to make your own words, before you start, select 'start new program' from the menu to discard any program present.

Call-up Notepad from the menu, press TAB to go into edit mode, and enter:

```
SCORE
24, 0:
GGe/G/ e/c/DE
F/d/G/ e/c/// ^
```

## Programming with AMPLE notation

The SCORE command sets all musical values, including ',' and ':' and many you haven't seen yet, to sensible starting values. You should put this at the start of each AMPLE notation tune. The next three lines hold our example tune, split up for convenience.

Before you can play this, you must have a voice with instrument, as before. In command mode, enter

```
1 VOICES Vibglock
```

Now return to edit mode and press f1 play to hear the tune. Once you've set up a voice, you don't need to do it again until you want to change the instrument. To play the tune at any time, just press f1.

When the cursor returns, you can try changing the tune. You could try replacing some of the notes by holds, or some of the holds by notes or rests, and so on. If you change every hold to a rest, the tune will play in a broken 'staccato' style.

## making into a word

To make a word containing the AMPLE music, you first set a name for it using the NAME command, and then make the word with the command MAKE, which can be on the same line. For example, to make the example above into a word, enter

```
"oranges"NAME MAKE
```

When the prompt returns, if you enter SHOW you will see the name of the new word on the list: if you started a new program, it will be the only one.

To get the word back for editing at a future time, you use the command EDIT:

```
"oranges"EDIT
```

This also sets the 'making' name to that of the word you edited, so to remake the same word with the changed contents, you just enter MAKE with no NAME beforehand.

## accidentals and key signatures

AMPLE's symbols for accidentals are, logically:

+ (sharp) and - (flat)

These go before the note, for example:

```
24, 0: C//D -E/c/ +D/// ^
           /      \
        E flat    D sharp
```

Unlike accidentals in sheet music, but like those in the Staff Editor, + and - apply to the next note only, so you must put them on each note that you want to be affected.

To set a key signature, you put a set of sharpened or flattened notes inside K( ... )K. The accidentals are remembered and automatically applied to all notes of the same letter name. Here are some examples:

```
K( +F )K          G major or E minor
K( -B -E )K       Bb major or G minor
```

The key signature goes after the SCORE, and before the first note.

AMPLE's natural symbol is '=' (equals) so to cancel the effect of the key signature on a note, you put '=' before it. Like + and -, this applies to the next note only.

## time signature and bars

You can include bar lines and the AMPLÉ equivalent of a time signature to automatically check for extra or missing beats in a bar. You use the word BAR with the number of beats per bar before it, and you put a length setting first to indicate the length of the beat. The 'l' character (top right of keyboard) is the bar line - it checks that there has been the correct number of beats since the last bar line. Here's an example:

```
SCORE  48, 3 BAR          % crotchet beat, 3 per bar
//                          % (first bar starts on beat 3)
0: C | CEG | C// | a// | // % ) phrase 1
    a | fGA | g// | /// | // % )
0: C | CEG | G// | d// | // % ) phrase 2
    E | Fed | c// | /// | // % )
    ^ |
```

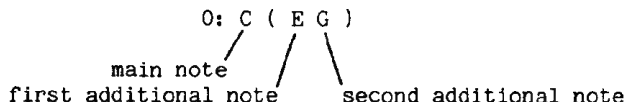
The time signature and bar lines are only for checking - they don't affect the sound.

This example illustrates some other points:

- \* long tunes are split into phrases that each start with an octave setting - better than a long tune with one setting at the start
- \* bar lines can fall in the middle of notes with no extra symbols needed - the AMPLE is much simpler than tied notes on the staff
- \* tunes can be laid out in phrases, even if, as here, bars are split over lines because the phrase starts in the middle of a bar. You should find it easy to follow the music as it plays

## chords

In AMPLE music notation, each chord is written as a main note plus up to seven additional notes which play at the same time. The additional notes go in round brackets after the main note, for example, here is a C major chord:



The pitches of the additional notes is just as if they were normal notes, so in this case the three notes are the middle C, and the E and G immediately above.

When it comes to putting chords together, each chord works exactly like a note: you simply string them together in sequence, using holds (after the whole chord) to extend the length. The pitch of each main note (and therefore the whole chord) is up or down from the previous main note - the additional notes do not count in the pitch. Here's an example that you should recognise when you hear it:

```
SCORE K( +F )K O:
48, G(BD) 16, G(BD) G(BD) G(BD)
48, e(BE) e(BE)
    d(BF) F(BD)
    e(BE) /
    ^ ( ^ ^ )
```

To play it, first set up enough voices with instruments for the chords, by entering the commands

3 VOICES Upright

Now return to edit mode and press f1 play to hear it.

Points to note in this example are:

- \* a 'chord of rests', ^(^), is used to end the sequence
- \* the notes of a chord can be close together in pitch or spread apart: the first chord is 'close', but the d(BF) is spread, due to the order of the pitches
- \* the length setting is changed to 16 to get the triplets at the start - three beats in the length of one ( $16 \times 3 = 48$ )

The chord of rests can become awkward if needed often, so there is an alternative - the 'chord rest' ^; which does rests on all voices.

You can mix notes, holds and rests in a single chord, if necessary. The hold makes the note on that voice hold on through the chord, and the rest silences that voice. If there is nothing for a voice, a hold is assumed - this lets you extend the whole chord with a single / , but means that when going from one chord to another with less notes, you must silence the unused voices with rests.

## slurs

AMPLE's slur symbol is the '~'. This goes before the note to be slurred, and has the effect, as on the staff, of joining its sound to that of the note before it. It works by disabling the striking of the envelopes, so the only change you hear is the pitch of the second note. If the envelope has died away by the time the slurred note starts, then you will not hear it at all.

In AMPLE, ties are totally redundant since a note written with holds can have bar lines, spaces etc. in the middle of it anyway. However if you use the slur symbol as a tie, that is, on a note which is the same pitch as the one before, you will get the expected effect.

If you are used to staff notation, then you may have some confused ideas about slurs, which we shall clear up here. Firstly, the slur symbol has absolutely no additional effect upon the previous note. It is a mistake to think of 'two slurred notes' - there is only one slurred note and that is the one after the symbol. Exactly the same applies to 'tied notes'. Secondly, you cannot play a slur on an instrument like a piano - it must have separate control of the envelope (for example, air flow or bow pressure) and the pitch (for example, fingering). The nearest thing to a slur on the piano is 'legato' (no gap between the note) and this is how a slur symbol in piano music is interpreted.

## AMPLE notation and staff notation

AMPLE notation is the 'native' music notation of the system, so even music entered in the Staff Editor can be displayed and edited in this form. You will have already seen that the Staff Editor displays the music in AMPLE notation when it is put into a word. If you take a score word that was created on the staff, you can TYPE it or GET it in to Notepad to see the AMPLE version. Try this on a staff word of your own or one of the 'part2' words in 'Manor'.

You will see what should by now be familiar AMPLE notation. All the staff symbols are represented using the small number of AMPLE symbols we have seen so far in this chapter. In particular, the following items correspond:

staff	AMPLE
clef	SCORE
end line	0,^ (zero-length rests)
tied note	~letter (slurred note of same pitch)

A tied note is represented as, for example, ~C/// rather than the obvious //// so that the tie is preserved when going back to the staff, even though this information is entirely redundant musically.

If you are interested, you can learn a great deal about the structure of music notation, above the confusing illogicalities of staff notation, by looking at the way staff and AMPLE notations correspond. For example, you will see that regular note values are multiples of a fundamental beat, and irregular values such as triplets and duplets represent a change in that beat, and not new values as such.

## marking staff words

When you start using AMPLE notation in earnest, you may find yourself writing things in natural ways which you then discover cannot be represented properly in staff notation. Some of the example pieces are written like this, even ones which were originally published in staff notation.

To make sure that such a word is not edited in the Staff Editor by mistake, all staff-editable words are marked with '%STAFF', as the very first thing on the first line. This does not show on the screen of the Staff Editor since it is removed by GET and replaced by MAKE. If you try to edit a non-STAFF word in the Staff Editor, it will come up empty, meaning there is no staff-editable music in it.

You don't have to think about this unless you are deliberately entering music in AMPLE for editing on the staff. If you are entering

a piece from a listing, you'd normally enter %STAFF where it appeared as a matter of course, like any other instruction.

This system is designed to suit the natural progression from using the staff to using AMPLE notation directly. If you have reached the stage of entering music in AMPLE notation, then presumably you don't want to take it back to the staff. If you do, and the music is not too complicated, you just add a %STAFF to the start.

## borrowed instructions

The Staff Editor's 'above-staff' instructions, such as 'Len' and ' (accent) are in fact AMPLE notation instructions, 'borrowed' for use on the staff. Any instruction you can put above the staff, you can use in AMPLE notation music. Here's a summary:

number Len	set length/gap between notes
Perc	use note pitches and hit voices
number =L	set dynamic level
'	accent next note
number 'L	set accent level
number @	set transposition

There are many further such instructions including true music words, and extensions such as special effects. Some you can also use above the staff, but others only work in the freer environment of AMPLE notation. These are introduced in the following sections, and are described in reference form in the chapters 'AMPLE music words' and 'Special effects'.

## echo

One of the most exciting special effects is echo. You can produce true echoes with full control over the volume, stereo position and even instrument selection of the original sound and each echo, because the effect uses separate voices on the player - the notes played on voice one are echoed on successive voices.

You turn on the echo effect with an 'Echo' instruction at the appropriate point in the music, usually the start of a section. 'Echo' takes two numbers, for example:

```
SCORE 48, 4 BAR K( +F )K 12 3 Echo
                        |  |
                        delay time    number of voices
```

The delay is the time between echoes, and is measured in timebase units, so this example uses a delay of one semiquaver. Specifying

three voices means that there will be two echoes in addition to the main sound. For experimentation, you set these voices up as you would for chords. Individual voice adjustments are made using the Mixing Desk.

Echo is a musical effect so it is cancelled by SCORE. You can turn it off manually with 'OFF Echo'.

## experimenting with echo

You can experiment with echo by typing notes in directly at the keyboard, and using the Mixing Desk to control the voices.

Call-up the Mixing Desk, and enter

```
READY
8 VOICES Vibglock
```

If you now switch to edit mode, you will see the eight voices ready to be mixed as if they were in a piece.

To play some echoes, enter

```
SCORE 48 2 Echo
C/// G^f-ed ^
```

Try increasing the number of echoes from 2 to 8, and changing the delay. You can enter single notes and rests followed by return to hear the effect.

To balance the echo voices, switch to the Mixing Desk's edit mode and adjust the volume faders and pan knobs as normal. The echo effect moves across the voices from 1 to 8, so 1 is the main sound. You could try making them successively quieter, and on alternate sides of the stereo field. You could even put a different instrument on each voice, but with any more than two echoes, this will probably sound muddled.

You can keep a direct mix such as this just as you would a normal one.



## using echo in a piece

You will find many ways to use echo in a piece, and if you have any voices spare, it is often worth using them up on echoes. Here are some suggestions:

- \* reverberation - one or two very short echoes on a quiet voice to give a reverberation effect
- \* slapback - a single short echo about as loud as the main voice, and often at the opposite pan position
- \* doubling - a short delay on a second voice with a slightly different or widely contrasting instrument
- \* harmonies - a second voice with a delay of exactly one or more beats on a carefully composed tune - a 'close canon'. This works well with arpeggios.
- \* rhythmic interleaving - one or more voices with dotted or triplet time values, adding interleaved beats. Very effective on sequencer-style patterns.
- \* repeating echoes - a long delay on three or more voices, giving a typical 'tape-loop' style echo

The echo voices can be adjusted with DETUNE and TRANS in addition to the Mixing Desk controls - see the chapter 'Building pieces' for information.

## percussion scores

AMPLE has a special music symbol for 'hits' - scored notes which have no pitches, as played on drum for example. This is 'X', and it works like a note in all other respects, for example

```
X/// X/X/ X//X XX/X
```

is a simple percussion score. The instrument plays at the last pitch set - if it is a purpose-designed percussion instrument, this is its pre-defined pitch, but it could just as well be the pitch of a previous note on the voice. (This can be useful for repeating notes - see 'X' in the chapter 'AMPLE music words' for details).

When it comes to multi-part percussion, you can use the 'Perc' effect as you do on the staff. The note letters D to B then play hits on successive voices:

```
D/E/ D/D(E)/ D/ED DDED
```

You could alternatively define your own percussion hit symbols for the

required voices, using ';' to set the voice:

```
"x" [ 1;X ] "y" [ 2;X ]  
...  
x/y/x(x(y)/ x/yx xxvx
```

## relative and gradual changes

Finally in this chapter, we come to gradual changes of dynamic level (for example, crescendo) and tempo (for example, rallentando) through the music. This is an advanced feature which is only introduced here, and dealt with in full in the reference chapter 'AMPLE music words'.

The words +L and -L work with =L to set gradual changes of dynamic level. +L takes two numbers that together specify the amount and length of the change, for example:

```
          30 8 +L  
          |  |  
amount of level change    length in number of beats
```

Note that the length is in beats, set by ',', not timebase units. This makes it much easier to specify the end position of the slope. If you want an change of level that is instantaneous, but relative to the =L setting, you use +L with a length of 0 beats.

-L is identical except that the level decreases.

The dynamic level is in fact a general-purpose musical level which, though used for volume by default, can be re-directed to control stereo position and pitch slide. See 'Autopan' and 'Slide' in the chapter 'Special effects' for details.

+T and -T produce relative and gradual changes of tempo, and work in the same fashion as +L and -L:

```
          10 16 +T  
          |  |  
amount of change    number of beats
```

Instantaneous changes (zero number of beats) are particularly useful since they are relative, so you can, for example double the tempo, or speed up slightly and then return to the original, without having to know the original tempo set on the Mixing Desk. You don't normally set the tempo to an absolute value (with =T) in the middle of music.

(Gradual tempo changes are deliberately cancelled by a new tempo setting in a mix. If you want a gradual change over more than one section, and then want to change the mix in the middle, you exclude the 'tempo' setting from the mix - see 'Building peices' for details).

## 7 Making instruments

The instruments we have used so far have been preset - supplied as standard with the system. You can easily make your own by either altering preset ones, or creating new ones from scratch.

To examine, change or create an instrument, we use Notepad in 'panel' mode. This provides a control panel rather like that of a synthesiser, with equivalents to 'knobs' and 'switches' for the variable parameters of the sound. Knobs take the form of numbers which you can increase or decrease, and switches can be set to ON or OFF. There are also 'word' items which you set from a menu of available options.

Rather than having a single unwieldy panel which includes every control, the panel layout varies to suit the class of instrument, leaving out those controls that are not relevant. You can edit the panel layout to change or add controls, or even design a custom panel to your requirements.

You hear the results of changes you make immediately, either on a voice you have set up just for editing, or on any voice in a piece of music playing at the same time.

### examining an instrument

You will know how to use Notepad for text editing from previous chapters. Apart from the editing itself, Notepad works in exactly the same way for editing instruments.

Before starting this session, save the existing program if you want to keep it, and select 'New program' from the menu (or enter the command NEW) to start afresh. When you get to building complete programs, you will usually call-up Notepad to edit an instrument in the existing program, so you won't clear the program.

To start this session, select 'Ready system' from the menu (or enter the command READY). This stops any program that is playing, frees voices and additional memory, and makes initial settings. (In future, you may want to enter Notepad with the program still running, or keep the voices you have set up, in which case you will not do a READY.)

Select 'Notepad' from the main menu. The Notepad screen will appear, leaving you in command mode. Remember that to switch between command and edit modes you press TAB, and to return to the main menu you press

## Making instruments

f0 in command mode.

To get an existing instrument for examination or editing, you use the GET command on the name of the instrument. We will start with the most basic preset one:

```
"Simpleins"GET
```

The contents of 'Simpleins' will appear at the top of the editing area, like this:

```
-----  
2 CHANS  
1 CHAN  
Hollow      Flat      Peaked  
-----  
|           |           |  
/           |           \  
waveform    pitch envelope  amplitude envelope
```

The first two lines are concerned with channels - for the moment, all we are interested in is that they provide one channel of sound, meaning that this instrument is a relatively simple one made up of a single tone.

The third line is the one we are interested in. It shows the waveform, pitch envelope and amplitude envelope selections - the settings that are responsible for the sound of the instrument. The waveform determines the tone quality - 'Hollow' gives a simple, hollow tone. The pitch envelope provides pitch variation during the note - in this case it is 'Flat', that is, there is no variation. The amplitude envelope determines the amplitude (or 'loudness') of the sound during the note - 'Peaked' gives a simple organ-like shape with a slight emphasis at the start.

## playing the instrument

In order to play the instrument and hear the effect of changes to it, you get a voice and put the instrument on it.

To get a voice, make sure you are in command mode and enter

```
1 VOICES
```

To put the instrument on the voice, go into edit mode (press TAB) and press f1 play (this executes the contents of the instrument, like putting an instrument name after the VOICES).

You can now hear the sound of the instrument by returning to edit mode

and pressing RETURN. This plays a note at middle C which lasts until you lift the RETURN key.

To try the instrument at different pitches, you can enter AMPLE notation note letters in command mode. Start by switching to command mode and entering

SCORE

to prepare for notes. You now enter single note letters (C,D,E,F,G,A and B) followed by RETURN to play different pitches, using upper-case letters to go up in pitch, lower-case letters to go down and ^ to stop the sound. You can first set the octave of the note by a number and ':' symbol, for example:

```
0:C      % middle C
-1:G     % a low G
```

Once you've settled on a pitch, return to edit mode and use RETURN to play notes at that pitch.

## changing settings

You may have noticed that the editing cursor is a block, rather than the usual line - this indicates that you are in Notepad's panel mode, the instrument-editing alternative to its usual text mode. Notepad selects panel mode automatically when you edit an instrument, so you don't need to worry about this.

To change the settings of the instrument, you use the edit mode cursor keys.

To move from item to item, you press the normal (without SHIFT) up, down, left and right keys. If you try them now, you will find that you can only move horizontally along the line. This is because the only variable items in 'Simpleins' (the waveform and envelopes) are on this line - the CHAN and CHANS settings are fixed.

To change a particular item, you move to it and press SHIFT. A menu of alternatives appears over the command area, with a cursor on the current selection. With the SHIFT key still down you use the cursor keys to move the cursor to the option you want, and then release SHIFT to select it. The new selection will appear on the screen in place of the old one.

## Making instruments

For example, to change the waveform, you

- \* move to the waveform word (the one on the left)
- \* press SHIFT
- \* while holding SHIFT, move to the new waveform option
- \* release SHIFT

If, after seeing the menu, you decide you want to keep the current selection, just release SHIFT without moving the cursor.

You can hear the effect of your change by pressing RETURN.

## trying out waveforms and envelopes

You could now go through all the waveforms and envelopes trying them out. We suggest you return each item to the original 'Simpleins' setting before varying another, to make sure you don't get confused between the effects of different items. In particular, use a straight-forward amplitude envelope like 'Peaked' or 'Onoff' when trying different pitch envelopes.

Your changes only take effect on the settings stored by the editor and on the sound - not on the instrument word itself. You can prove this by re-entering '1 VOICES Simpleins' - the sound returns to the original. The screen settings are unaffected, so you can restore the sound to match the screen settings by pressing f1 play in edit mode.

At any time, you can get back to the original settings with:

```
"Simpleins"GET
```

Detailed descriptions of all the waveforms and envelopes appear in the chapter 'Sound Instructions'.

## making into a word

The set of items on the screen constitutes a new instrument which you can make into a new word, allowing you to

- \* call the instrument up by name
- \* use the instrument in a piece
- \* save the instrument as part of a complete program

The top area of the screen shows the name of the instrument that you used originally, and the name that will be given to the new word you make. This is currently 'newword', but you can set it to a name of your choice with the NAME command. For example, in command mode enter

"myins"NAME

Now, to make the instrument word under the name 'myins', you enter

MAKE

If you now use SHOW to display the list of user word names, you will find 'myins' in it.

You can save your instrument word, along with any other user words that are present, as a program, either with the 'Save program' option on the main menu, or by the direct SAVE command:

"instrus"SAVE

To reload the program, use 'Load program' from the menu, or a direct LOAD command:

"instrus"LOAD

The chapter 'Building pieces' describes how to add your saved instrument words to an existing program.

## modifying existing instruments

You cannot change the contents of a preset word, so if you want a modified version of a preset instrument, you make it as a new word of your own. Preset words have upper-case initials and your word names must only use lower-case letters, so there should never be confusion between the two.

Once you have made an instrument as a word, you can recall it for editing using GET as you would for a preset instrument. In this case, GET sets the 'making' name to that of the original word, so that to replace the original word by a new version, you don't need to set the name with NAME.

## trying out different notes

The 'trying' line at the top of the screen shows the commands that are carried out when you press RETURN to try the sound. At the moment it should say 'RETGATE', the name of the command that plays the sound until RETURN is lifted.

## Making instruments

You can set your own 'trying' line to play a sequence of notes using the TRY command, for example:

```
"0:CDEFG^"TRY SCORE
```

The '0:' sets the octave and the '^' symbol silences the last note. 'SCORE' is a separate command that prepares for notes as before. Now, when you return to edit mode and press RETURN, the five-note scale will play.

You can try the instrument on more than one voice simultaneously, to hear how it sounds when playing chords. You enter, for example:

```
3 VOICES
```

and then put the instrument on all these voices by pressing f1 play in edit mode as before. To play chords, you use note letters followed by the letters for the simultaneous notes in brackets, for example:

```
0:C(EG) D(FA) ^;
```

The '^;' symbol silences all the voices. To put this on the trying line, enter:

```
"0:C(EG) D(FA) ^;"TRY SCORE
```

## two-channel instruments

The 'Simpleins' instrument uses only one channel of sound, but most instruments use two, enormously expanding the range of sounds. A channel pair is used in three main ways:

- \* with slightly different settings to enrich the sound
- \* with contrasting settings to produce a complex varying sound
- \* in combination with modulation, producing totally new sounds

We'll start with a look at an example of the first; the synthetic piano sound 'Upright'. Enter

```
"Upright"GET
```



The lines of the instrument are as follows:

```

-----
2 CHANS                                number of channels
2 CHAN                                select channel 2
    0 SHIFT    200 OFFSET 128 AMP      instructions for ch. 2
EVERY CHAN                                select channels 1 and 2
    Round      Flat      Strike      instructions for
    ON PHSET                                channels 1 and 2
-----

```

You will recognise the line starting 'Round' as the waveform, pitch envelope and amplitude envelope selections. The most distinctive of these is the amplitude envelope 'Strike', which has an initial peak very characteristic of the piano. The 'EVERY CHAN' on the line above means that these selections are for both of the channels, so the two channels will always have the same waveform and envelopes.

The other long line has three number settings which apply to channel 2 only, indicated by the '2 CHAN' above it. These settings determine the differences between the two channels that give the richer sound.

To listen to this instrument, set it up on a voice as previously.

## adjusting numbers and flags

The SHIFT cursor keys let you change numbers in the same way as waveforms and envelopes, except there is no menu - the keys increase and decrease the number in place on the screen, like adjusting a knob:

<u>SHIFT up</u>	increase by one
<u>SHIFT down</u>	decrease by one
<u>SHIFT left</u>	increase by ten
<u>SHIFT right</u>	decrease by ten

For example, to change the OFFSET value, you move the cursor to it and press SHIFT up to increase by one, and SHIFT down to decrease by one. You can set any number to zero by pressing '&' (SHIFT underline, just next to the cursor keys).

To change a flag (a 'switch' value that is ON or OFF), you move the cursor to it and press SHIFT up to turn it ON, and SHIFT down to turn it OFF.

## offset

The OFFSET instruction makes the channel slightly sharp or flat. It is commonly used to make two channels very slightly 'out-of-tune' with each other. In 'Upright', the two channels are used for the effect of two strings on each key. The offset makes one string slightly 'flat' or 'sharp'.

Try adjusting the offset, and listen to the following settings:

- 0 no offset: flat, less interesting sound
- 200 normal offset: warm natural sound of slightly detuned strings
- 600 large offset: out-of-tune 'honky-tonk' piano sound

## amplitude

The AMP instruction sets the amplitude in the range 0 (off) to 128 (loudest). Channel 1 is at the loudest setting by default, so AMP is mainly used for setting the balance of channel 2. In 'Upright', the two strings are the same loudness, since channel 2's AMP setting is at maximum.

Try setting channel 2's AMP to zero (move to it and press pound, 'f'), to remove the sound of the second string. If you set it to 100 AMP, you will hear a faint two-string effect.

## shift

The SHIFT instruction raises or lowers the pitch of the channel so it plays a fixed interval with another channel. (Don't confuse the SHIFT instruction with the SHIFT key). In 'Upright' the instruction is '0 SHIFT' so the two strings are at the same pitch (though detuned by OFFSET). 0 SHIFT is the default, so in fact the SHIFT instruction has no effect in the unmodified 'Upright' instrument - it is included just so it can be changed.

Try raising the SHIFT value to make the two strings play an interval apart. 192 SHIFT produces an octave - this can have the effect of either two sounds an octave apart or one new sound, depending on the context.

## phase setting

The ON PHSET instruction gives a precise hard start to the sound waveform at the beginning of each note, as opposed to a soft continuation of the previous note. In 'Upright', it contributes to the hammer strike effect. Try switching it to OFF (by moving to it and pressing SHIFT down) to discover its effect.

## modifications and new instruments

Here are some further modifications to 'Upright' for you to try one-at-a-time and in combination. Remember that you can recall the original 'Upright' at any time using GET.

- \* change the waveform 'Round' to 'Hollow' or 'Clear'
- \* change the pitch envelope 'Flat' to 'Vibrato' or 'Delvib'
- \* change the amplitude envelope 'Strike' to 'Short' or 'Reverse'

You can easily make completely new instruments using the same panel design as 'Upright', starting from the same settings. For example, to create a church organ sound, make the following changes:

- \* 192 SHIFT
- \* waveform to 'Pipes'
- \* amplitude to 'Onoff'

You will end up with an instrument similar to the preset 'Organ'.

## contrasting channels

'Vibglock' is an instrument that has a glockenspiel-like sound with a gentle vibrato as it dies away. It is an example of one whose sound has two different parts, so the channels have contrasting settings.

Enter

```
"Vibglock"GET
```

Here's what appears:

```
-----
2 CHANS
1 CHAN
  Clear      Delvib      Long      instructions for channel 1
2 CHAN
  High       Flat       Short     instructions
128 AMP                               for channel 2
EVERY CHAN
  ON PHSET                               instruction for ch. 1 and 2
-----
```

Channel 1 makes the main sound - this dies away slowly ('Long') with a vibrato at the end ('Delvib'). Channel 2 makes the initial 'clonk' of the mallet strike - it has a special waveform ('High') and dies away quickly ('Short'). The ON PHSET creates a precise strike.

The waveforms and envelopes are carefully selected so that the two component sounds combine to make a complex single sound. You can vary the brightness of the strike using channel 2's AMP setting. Reducing this to 100 makes a duller sound.

## a drum instrument

The instrument 'Drum' has a similar panel to 'Vibglock', but an entirely different sound:

```
-----
2 CHANS
1 CHAN
  Pure      Zap      Burst      instructions
-38 PITCH                               for channel 1
2 CHAN
  Metal     Flat     Click     instructions
-20 PITCH  128 AMP    for channel 2
EVERY CHAN
  OFF PSNS  ON PHSET    instructions for ch. 1 & 2
-----
```

The sound is that of a modern drum-machine-style bass drum. Channel 1 makes the main tone of the drum skin. It has an extreme pitch sweep on it ('Zap') which in this short sound makes a more complex skin sound. Channel 2 produces the all-important thwack of the beater hitting the skin, by using a rough waveform ('Metal') and an envelope ('Click') that is so short that only one cycle of the waveform is let through.

Try reducing the AMP setting to show the contribution of channel 2.

Being a drum, the instrument has pitch of its own as set for each channel by PITCH. The OFF PSENS instruction makes both channels ignore further pitch settings, so the pitch of notes has no effect.

You can think of the PITCH of the thwack as setting the replay speed of a very short recording ('sample') of a strike sound. If you first switch PSENS to ON, you can lengthen and lower the thwack by decreasing the PITCH setting. Channel 1's PITCH independently controls the tightness of the drum skin.

## modulation

Another way of combining channels is by modulation. This transforms the sound of channel 1 by feeding channel 2 into it in various ways. The waveforms of both channels, the offset and the shift all contribute to the final sound. Channel 2's sound is unaffected and can be added in separately. Using pitch envelopes, the sound can be made to change through the note.

There are three types of modulation available: ring modulation, synchronisation and frequency modulation.

## ring modulation

Ring modulation makes a complex hybrid of the two channels' waveforms to produce sounds that are richer, brighter and at times rough and distorted. The pitch interval and waveform selections are both important to the final effect.

## Making instruments

As an example, we will look at 'Ringsyn':

```
-----  
2 CHANS  
1 CHAN  
  ON RM  
2 CHAN  
  0 SHIFT    150 OFFSET 128 AMP  
EVERY CHAN  
  Bright      Delvib      Verylong  
  ON PHSET  
-----
```

Switching RM to OFF will show you the effect of the modulation. To hear the modulation effect alone, set channel 2's AMP to zero.

The effect of an offset is much greater with ring modulation. Here it creates the throbbing effect: more offset gives faster throbbing. You sometimes see ON PHSET used to restart the throb cycle on each note.

You can slow the throb down to create a steadily-changing tone. For example, these changes make the sound of a swooping aeroplane:

- \* 20 OFFSET (for a changing tone)
- \* 'Delvib' to 'Wide' (for a swooping pitch)

To set this to a suitable pitch, play a 0:C on it before using RETURN to play the sound. (If you have changed the trying line to play some notes, enter "RETGATE"TRY to return to the simple RETURN key effect.)

Setting SHIFT for different musical intervals produces different modulated tones: try 112, 192 and 384, after returning the pitch envelope to 'Delvib'. Discordant intervals produce dense non-harmonic tones which are useful for rough sound effects, for example, a racing-car engine:

- \* 45 SHIFT
- \* 'Delvib' to 'Wide'
- \* 'Verylong' to 'Onoff'

The most effective pitch is around -1:G (the G below middle C). To play the sound of eight cars together, first set up eight voices:

```
8 VOICES
TAB f1 play TAB .
```

Then, be ready to turn down the amplifier volume, and enter:

```
SCORE -2:c(100,FBEGDAG)
```

Press ESCAPE for silence.

## noise sounds

Ring modulation of a complex wave is one way of making random noise. The best waveform for this is 'Metal', which is itself random. The instrument 'Cymbal' is a good example:

```
-----
2 CHANS
1 CHAN
  ON RM                100 AMP
2 CHAN
  110 SHIFT  -20 OFFSET 128 AMP
EVERY CHAN
  Metal      Flat      Long
  24 PITCH   OFF PSENS  ON PHSET
-----
```

Both channels use the 'Metal' waveform so while channel 1 produces the noise part of the cymbal sound, channel 2 adds the metallic ring. Channel 1's AMP instruction reduces the level of the noise part.

This instrument is the basis for many interesting metallic sounds. A good start is to lower the pitch and set 0 SHIFT, adjusting OFFSET to taste. The result is just as interesting without ring modulation.

## synchronisation

Synchronisation (or 'sync' for short) produces coloured (for example, vowel-like) sounds by forcing channel 1 to play the pitch of channel 2, leaving channel 1's original pitch to 'sound through' as a kind of resonance.

The pitch of the note is always that of channel 2. Channel 1's pitch controls the tone, and its pitch envelope changes the tone dynamically.

## Making instruments

The example instrument is 'Wha', a human voice sound:

```
-----  
2 CHANS  
1 CHAN  
  Hollow      Wow      Soft  
  150 SHIFT   ON SYNC  
2 CHAN  
  Syncer      Deepvib   Peaked  
EVERY CHAN  
-----
```

Channel 2 uses a special waveform ('Syncer') that gives the strongest sync effect, and a pitch envelope ('Deepvib') for a voice-like vibrato. Channel 1 has the ON SYNC instruction, a SHIFT to set the basic tone, and a pitch envelope for a 'waow'-like up-and-down sweep at the start of each note.

Notice that there is an 'EVERY CHAN' line even though there are no instructions under it. The last channels selection should always be EVERY CHAN so that following settings automatically apply to all channels.

Try this instrument at different pitches to get 'big' and 'little' people's voices. The SHIFT value determines how 'high' the resonance tone is: increasing the SHIFT gives the effect of a wider mouth.

The ON SYNC instruction has the side effect of ignoring any further changes to the pitch envelope on channel 1, so that the real pitch envelope (channel 2's) can easily be changed outside the instrument without disturbing the sync effect's pitch envelope. To change channel 1's envelope for a different tone shape, set SYNC to OFF first, for example:

```
* set OFF SYNC  
* change 'Wow' to 'Drop'  
* set ON SYNC
```

## frequency modulation

Frequency modulation ('FM') varies the frequency of channel 1 so fast that new waveforms are created. The variation is provided by channel 2 so, like ring modulation, the pitch interval strongly affects the sound. FM is most used for non-harmonic sounds like bells and chimes.



A good example is 'Yakbell', a small metal bell instrument:

```
-----
2 CHANS
1 CHAN
  90 SHIFT   80 FM
EVERY CHAN
  Pure      Flat      Percuss
  26 PITCH  ON PHSET
-----
```

Unlike ring modulation and sync, FM is a variable-depth effect - the FM value can range from 0 (off) to 255 (maximum). Channel 1 has the FM instruction and a SHIFT to set the pitch interval. Channel 2 is just used for the modulation - it has no AMP, so it doesn't sound. Both channels have a pure basic waveform ('Pure').

The intended, natural pitch of 'Yakbell' is 2:D. If you play the instrument with the X letter instead of note letters, this pitch will be used automatically.

Try varying the SHIFT setting for a variety of 'clangy' sounds. You will find that positive values give the strongest effects, and, once more, concordant musical intervals are special. Try increasing the FM depth for a stronger sound.

This simple instrument panel is very versatile - the unmodified instrument varies from the original bell at high pitches, through sounds like steel drums in the middle and timpani at the bottom. You can adjust the SHIFT, FM and envelope settings to fine-tune any of these variations.

## stereo instruments

A stereo instrument is one that has channels at different stereo positions. Where the channels have similar sounds with an offset, this spreads the sound out and adds extra ambience. You can use different envelopes on the two channels to make a single sound that sweeps across, or simply put separate sound components on opposite sides for effect.

## Making instruments

'Panflute' is an 'ambient' stereo instrument:

```
-----  
2 CHANS  
1 CHAN  
  
-3 POS  
2 CHAN  
500 OFFSET 128 AMP 3 POS  
EVERY CHAN  
Pure Delvib Puff  
-----
```

POS sets the position, in the range -3 (left) to 3 (right) with 0 being centre. The channel positions are balanced so that the overall position is normally central. To decrease the spread, just move both channels' POS values towards zero.

Stereo instruments can be positioned on the Mixing Desk just like normal ones, and the pan control compensates for any channel which hits full left or full right.

## adjusting while playing

You can adjust the sound of any voice of a piece of music while it is playing, in the same way as you adjust the sound of a voice set up for this purpose. We do this by using the SHARE command to select a player, instead of setting up a voice with VOICES. SHARE takes the number of the player - all sound instructions from the editor (and those entered directly) are then sent to all voices of the numbered player.

To make adjustments while a piece of music is playing, first choose the player whose instrument you want to adjust, and find out the name of the instrument (the easiest way to do this is with the Mixing Desk).

Then carry out the following steps:

- \* enter RUN to start the piece playing
- \* select the player with SHARE
- \* get the instrument into Notepad with GET
- \* enter edit mode and make adjustments as usual

To illustrate, we will use the example piece 'To the Manor Born', and prepare to make adjustments on the voice that plays the tune - player 2, instrument 'brassy' (not preset, but part of the program).

- \* return to the menu (f0) and load the program 'manor'
- \* select 'Notepad' from the menu
- \* enter RUN
- \* enter 2 SHARE to select player 2
- \* enter "brassy"GET
- \* press TAB to enter edit mode

As an alternative to adjusting the existing instrument on a voice, you can edit a different instrument and press f1 play to put it on the selected voice.

Where a player has more than one voice, SHARE will normally select all of them, so instruments and adjustments will take effect on all voices. You can select just one of them using the command VOICE after SHARE. For example, in 'Manor', player 3 plays three-voice chords on an organ-type instrument.

3 SHARE 3 VOICE

will select the third (top) note of the organ.

You may find it easier to edit an instrument with a 'background' piece playing the sound continuously, rather than use RETURN to play the trying line. You can easily make a short repeating sequence of long, short, high, and low notes, and put it in a test piece for this purpose.

## modifying and designing panels

You can modify the current panel at any time using Notepad's text editing facilities to delete, replace and enter items. This you do in Notepad's text mode, - you switch between this and panel mode by pressing f2 text/panel. In text mode, the cursor is a line, and in panel mode it is a block.

Notepad's text mode is described in the chapter 'Editing words in text form'. Using it, you can:

- \* add an item to the end of an existing line of items
- \* add a new line of items below a channel selection
- \* add a new channel selection and lines of items below it
- \* replace an existing item with a new one
- \* remove an existing item, to save space in the instrument word
- \* remove a channel selection and the items below it, to save space
- \* enter a new panel layout, starting with the command CLEAR

## Making instruments

You will find that there is very little restriction on where you put items etc, but it is a good idea to follow these rules of instrument panel layout:

- \* the first line contains the CHANS instruction only
- \* each subsequent group of lines consists of a fixed line with the channel selection (1, 2, or EVERY CHAN), followed by variable lines for that selection
- \* each variable line has a number of items, separated by spaces
- \* each item is either a waveform or envelope word, or an instruction with a number or flag before it.
- \* each variable item has space allowed for its biggest required value. Waveform and envelope words are abbreviated to fit with a dot, and numbers and flags are limited to values that will fit.
- \* for clarity, the waveform, pitch envelope and amplitude envelope appear in that order on one line
- \* items are positioned approximately in columns so that vertical cursor movements move near-straight up and down between items

To save memory, you can leave out the waveform and/or envelope selection for a channel, whereupon the following defaults will be used:

```
waveform:      Hollow
pitch envelope: Flat
amplitude envelope: Peaked
```

Remember that just like any other setting, you will not be able to use panel mode to change a selection if it does not appear on the panel.

To enter an instrument from a printed page or program listing, you start with a CLEAR and then type the lines in one-by-one, including the spaces to lay the items out across the screen. When you've finished, check the screen against the page, set the name to that given at the top of the printed instrument, and make the word.

## continuation instruments

You often find that you need a slightly modified version of an existing instrument, for example, one with added stereo positioning, or just one envelope changed. As a more-efficient alternative to making a complete instrument with just the one difference, you can make a 'continuation' instrument; one which calls up an existing instrument and then applies further sound instructions to the voice.

The panel of a continuation instrument has the name of the original instrument plus the instruction ON CHANS in place of the usual 2 CHANS, followed by channel selections and sound instructions as normal. As an

example, here's a stereo variation of 'Moog':

```

-----
Moog ON CHANS                                original instrument
1 CHAN
-3 POS                                       further sound instructions
2 CHAN
3 POS
EVERY CHAN
-----

```

## many-channel instruments

So far we have seen instruments that use one or two channels of the 16 available, and this is enough for most sounds. For even more powerful and complex sounds, you can use any number of channels in pairs up to 16 in a single instrument. The total limit of 16 channels still applies, so when using instruments with more than two channels, the number of voices is reduced accordingly. For example, a single four-channel voice counts as two two-channel voices.

Many-channel instruments look very similar to two-channel instruments, but additional multiple channel specifiers may be used:

number PAIR CHAN	the pair containing the numbered channel, for example, 1 PAIR CHAN selects channels 1 & 2
ODD CHAN	all odd-numbered channels
EVEN CHAN	all even-numbered channels

Like EVERY CHAN, these allow settings common to more than one channel to appear just once, allowing easier editing and saving memory space.

## Making instruments

The following example is 'hotbass4', a four-channel development of the two-channel preset instrument 'Slapbass' - for each channel in 'Slapbass', 'hotbass4' uses a pair of channels to give a more powerful sound. This is not a preset instrument (indicated by the lower-case initial), so you must enter it yourself. Enter the command CLEAR, switch to edit mode, and enter the lines one-by one from the top, roughly copying the positioning of the items.

```
hotbass4
-----
4 CHANS
1 CHAN
  Hollow      Drop      Long
ON SYNC      100 SHIFT
2 CHAN
  Syncer      Flat      Peaked
3 PAIR CHAN
  Metal       Flat      Short
4 CHAN
  20 OFFSET
EVERY CHAN `
  128 AMP
-----
```

Channels 1 and 2 are combined with sync to make a strong main tone which starts bright and gets duller, while channels 3 and 4 make a metallic clank with a steady phasing sound resulting from the small offset. Listen to a long line of short notes to hear this effect.

The name has a '4' as the last character to indicate that it is a four-channel instrument - in the interests of clarity, you should stick to this convention in your own instruments.

## 8 Building pieces

So far we have seen how to create and edit the three main types of word:

score, instrument, mix

now put them together to make a complete piece

### simple pieces

There are three stages to making a simple multi-part piece: scoring, titling and mixing.

At the scoring stage, you enter the music for each part in either staff notation or AMPLE music notation. Before you start, you should plan out the score, and in particular how many parts there are and how many voices each one has. Here's a typical plan:

part 1	lead	1 voice
part 2	chords	4 voices
part 3	bass	1 voice
part 4	drums	2 voices

Remember that there is a total of eight voices.

You then enter the parts as individual words called 'part1', 'part2', 'part3' and so on. These are key names which the system makes reference to when combining the parts. (You don't have to put all the music for a part into this single word - unless the piece is small, it will be best to divide it into sub-sections. This is described later in this chapter).

At the end of this, you should have a 'part' word for each part in the piece.

### titling

The next stage is to create the RUN word for the piece. Every complete piece has a word RUN which displays the title and plays the piece from its component parts. (RUN is the exception to the rule of using lower-case letters for user word names - it deliberately looks

## Building pieces

like a system word.)

To create the RUN word, you use 'Notepad'. The RUN word usually contains:

- \* the word DISPLAY, to display the following lines starting with %
- \* up to 9 title lines, starting with %
- \* a PLAY instruction

You can use the markers in the Notepad top and bottom rules to centre the title lines on the screen.

The PLAY instruction plays the piece, using a specified list of players - for each part in the piece, you need the player of the same number. The list of player numbers goes in a string (that is, enclosed in double quotes), before the word PLAY, for example:

```
"1234"PLAY
```

A complete title word might look like this (including the Notepad rules):

```
-----|-----  
DISPLAY  
%      Chariots of Fire  
%      by  
%      Vangelis  
%  
%      programmed by  
%      Julian Reed  
%  
"123"PLAY  
-----|-----
```

## setting up a new mix

The final stage is to set up a new mix, which you do in the Mixing Desk. At this point you make reference to your piece plan to find out the number of voices on each part, and you enter this information using the pair of words SHARE and VOICES, as follows:

```
partnumber SHARE voicesnumber VOICES
```

The full sequence of SHARE-VOICES pairs needs a READY at the start, to discard any existing voices, and a CLEAR at the end to clear the desk



and set up the required voices. Here's a set-up sequence for a four-part piece:

```
READY
1 SHARE 1 VOICES
2 SHARE 4 VOICES
3 SHARE 1 VOICES
4 SHARE 2 VOICES
CLEAR
```

If you now go into edit mode, you will see each voice in place with 'Simpleins', the simple default instrument, selected. Any voice that you are not using is shown with brackets around its voice number, indicating that its settings are not included.

You can now work across the Desk choosing instruments and setting controls, either in silence or after pressing r to start the music playing. All existing voices are cleared when a piece starts, so if you set the voices up in silence, don't play the piece until the mix is made as a word. You can also set the tempo and tune at this point.

To store the complete mix, enter MAKE as usual. The piece will now use this mix each time it starts.

Save the final piece.

## leaving out players

You can run the piece with one or more players left out by simply leaving their numbers out of the PLAY string. This is particularly useful for trying out single parts, because PLAY can be entered as a direct command, for example:

```
"2"PLAY           % play part 2 only
```

## adding standard words

You can define standard words of your own, such as instruments and dynamic level words, and save them in text form so that they can be added to an existing program when making a piece. To save all the words in the program as text, enter:

```
*SPOOL text
WRITE
*SPOOL
```

## Building pieces

To add them to the program in memory at a later date, enter:

```
*EXEC text
```

## sections

Many pieces are naturally divided into large sections, for example, verses and choruses in songs, and often the same section may be used more than once at different points (a chorus for example).

You take advantage of this by entering each part as a number of separate section words. You can then make PLAY collect up the different sections for the parts and play them in a specified order.

You create your part section words under the names 'part1a', 'part2a' etc., and use the section suffix to call-up the whole section, in all parts, in a specified sequence. The sequence of section letters goes in the PLAY string, after the player numbers, with a dash before it, for example:

```
"123-aabb"PLAY
```

This plays section 'a' twice, then section 'b' twice.

There are many advantages to breaking a piece into sections. The obvious ones are that repeated sections are not duplicated and sections can be reordered at a stroke; a boon for composers and arrangers. You could try re-arranging the sections of 'Manor' in this way. Try creating a shorter version for a TV program title slot, for example.

## stopping and skipping

You can use special 'control' sections to include Mixing Desk session control settings in the sequence. This lets you program pauses at critical points, released from the Mixing Desk, and fast-winds to skip through sections you don't want for the moment. To avoid getting in the way of sound-playing voices, you should use part number 0 ( in fact player number 10) for the controls. Here are the definitions of the control sections:

```
"part0p" [ ON PAUSE ]      % pause
"part0s" [ ON FAST ]       % run at top speed
"part0r" [ OFF FAST ]      % return to normal speed
```

Here's a section that uses all three:

```

        pause until command
        |
    "12345-pabsabrcaba" PLAY
        | |
        | skip this sequence

```

When the piece plays, the control changes appear on the Mixing Desk just as if you had pressed the keys. This example sets ON PAUSE on the Mixing Desk immediately it starts, waiting for you to hit the space bar to start the music.

You don't need to be in the Mixing Desk to use the session controls from a piece. If you are in another editor, you can release the pause by entering the command

```
OFF PAUSE
```

or make it up as a command of your own:

```
"go" [ OFF PAUSE ]
```

## sub-sections

Musical sections are often divided into sub-sections, which, though they don't have the status of full sections, are still individual units which are useful to program as such. Some examples include short repeats, first and second time bars, and patterns in sequences.

AMPLE's word-defining ability is tailor-made for creating sub-sections and building them into full sections. Each sub-section is made as an individual word, and the full section word just calls them up in the required sequence. You are free to use a single sub-section in different sections and different parts. The section word is entered in Notepad.

The main difference between a section word and a sub-section word is that sub-section words do not start with a clef (on the staff) or a SCORE instruction (in AMPLE notation). The SCORE setting is put at the start of the section word, allowing settings such Len, Echo and transposition to be made in the section word, and still apply through the sub-section words. Here's an example, in Notepad:

```

-----
SCORE -4 Len
sub1 sub2 sub3 sub4
0,^
-----

```

## Building pieces

Sub-section words also don't need end lines (on the staff) and terminating rests (in AMPLE) - these should go at the end of the section word.

You can choose any legal names for your sub-section words. A suggested scheme is based on the part and section, with a further suffix for the sub-section, for example, 's1a1' : part1, section 'a', sub-section 1. Where appropriate, you can use more meaningful names such as 'part1afirst' and 'rhyfill'.

One particular new instruction that is useful with sub-sections is the loop, FOR( ... )FOR. The FOR( takes a number, and does all the words up to the )FOR that many times. This is an 'inside-word only' word, so you cannot play it direct from Notepad with f1.

What you use sub-sections for is largely up to you, but here are some suggestions:

application	section word contents
splitting up into music pages	page1 page2 page3
making up tunes from phrases	phr1 phr1 phr2 phr2 phr1 phr3
many repeats	7 FOR( tune1 )FOR tune8
first-time second-time	main1 first main1 second main2
rhythm patterns	pa1 pa1 pa1 pa2 pa1 pa1 pa1 pa3
transposed phrases	0@ riff 4@ riff 2@ riff 5@ riff

## changing mixes

The PLAY list allows you to include mix indicators to call-up new mixes between sections. The mixes are named 'mix1', 'mix2' etc, and you call them up using just their number digits, for example:

```
"1234-1a2a3a" PLAY      % play 'a' three times with mixes 1, 2 and 3
```

If you do have more than one mix in a piece, name them all with digits rather than leaving the one called 'mix'.

## sub-mixes

Sub-mixes are sub-sections of full mixes, and are used to change some parts of a mix while leaving the rest the same. They are commonly used to change instruments on just a few of the voices, or change level and pan while leaving the instruments the same. Unless you are really changing the whole mix, using full mixes in the piece is very inefficient and you should use a sub-mix instead.

Sub-mixes are very easy to create: you start with the full mix that it is based on, and use the 'new mix' command

M5MIX

which puts all parameters on the screen in brackets to show that they are excluded from the mix to be made by MAKE. You then adjust each of the parameters that you want to be different - the brackets automatically disappear to include them in the mix. You can individually exclude whole voices, whole parameters across all voices, and the tempo and tune settings.

To make the word, you set the name and enter MAKE as normal.

You can also remove an item from the sub-mix, that is, set its brackets again. To set the brackets on a voice, group of voices, tempo or tune, you move the cursor to the item and press 'B'. To change brackets on one of the 'horizontal' parameters, you use

```
I  instruments on all voices
P  pan on all voices
V  volume on all voices
```

To call-up a sub-mix, you include its number digit in the PLAY list just as for a mix. For example, if we had a full mix '1', and two sub mixes '2' and '3', we might use them in a PLAY list like this:

```
"1234-12ab3ab2ac"PLAY
```

In this case, the sub-mixes are being used to modify a single full mix, but you can just as well apply the same sub-mixes to many different full mixes.

Try out some sub-mixes on 'Manor'. The original has no mix changes in the music but there are convenient section breaks where you could add them.

## mix displays

The Mixing Desk display keeps up with the changes of mix in the piece, even to the extent of changing the name and bracket settings. This is very useful because it lets you stop the music at any point and MAKE a replacement mix, without having to worry about where it is called up and whether it is a full or sub mix.

The 'live' screen can also be a danger, since it may interrupt you in editing a mix. When composing mixes live on a piece which changes mixes, you should exit to command mode to save the new mix in good time (a temporary copy is made for MAKE on entering command mode, so

## Building pieces

further changes have no effect on the made word).

A better alternative is to make up a small PLAY list with the section you need repeated a few times, with no mix changes, for example:

```
"1234-1aaaaaaaaaaaaa"PLAY
```

## unmixes

With a bit of planning, you can transfer voices from player to player between sections, to make fuller use of the eight voices available. You free voices from players using an 'unmix' - a bit like a normal mix, except that instead of putting an instrument on a voice, it takes the instrument away. If you follow a mix with an unmix that matches it voice-for-voice, then all instruments will be removed, leaving all eight voices free. If you follow it with a sub-unmix for, say, just one player's voices, then these will be freed for use on other players in an immediately-following mix.

To make an unmix, you set the Desk exactly as you would for a mix, and then use the command

```
UMAKE
```

to make an unmix of the current name. Almost always, this will be a sub-unmix, so you will set the brackets on all voices that are to be unaffected. (Only the voice brackets mean anything in this case).

Here's an example mix sequence (shown going down the page) that transfers three voices from player 2 to players 3 to 5.

name	type	player-voice numbers			
mix1	mix	1-1	2-1	2-2	2-3
	...				
mix2	unmix	( )	2-1	2-2	2-3
	(result)	1-1	X-X	X-X	X-X
mix3	mix	1-1	3-1	4-1	5-1

By their very nature, unmixes are uneditable - if you need to change an unmix, you re-make it from the mix it is required to match.

**voice sound commands**

There are two voice sound commands for use inside section words, usually at the very start. These are sound commands, like those in instruments but applying to all channels on a voice. They are not cancelled by a clef or SCORE, but by the next instrument change on the voice.

The first is DETUNE, which adds a frequency offset to detune the voice relative to another (It is equivalent to but independent of OFFSET). It takes the amount of detuning as a number before it, in units of 0.0056Hz. A value of 50-200 is common.

You often use DETUNE where two voices may play the same pitch; in crossing melodic lines for example. The slight frequency difference ensures that the sounds of the two notes don't cancel each other due to phase effects, producing a 'drop-out' effect. DETUNE is also used to enrich the effect of Echo by detuning each of the voices slightly.

The DETUNE instruction always has a VOICE instruction before it to specify the voice, for example:

```
2 VOICE 100 DETUNE % detune voice 2
```

The second voice sound command is TRANS, which transposes the pitch of the voice specified by VOICE. Unlike the music word '@', this is a voice specific instruction, so it is mostly used for moving individual voices into different registers. It takes the number of semitones transposition, for example:

```
1 VOICE 12 TRANS % transpose up an octave
```





## 9 Further use of AMPLE

This chapter describes further features of the system for more advanced users, including details of how to adapt the system for use on non-standard computers.

### making menus

You can easily make your own menus like the Main Menu, for example, a 'jukebox' menu that plays pieces from disc. You just create a single menu word, using Notepad to lay out the screen display and add action commands for each option, and AMPLE does the rest: presenting the menu, controlling the cursor, and executing the selected option.

Here's an example 'jukebox' menu:

Debussy Piano Pieces	title line
Clair de Lune	)
Golliwog's Cake Walk	) options
...	)

-----

The menu word for this is as follows (with Notepad's rules at top and bottom):

-----  
MENUDISP  
% Debussy Piano Pieces  
%  
%Clair de Lune%"clair"LOAD RUN  
%Golliwog's Cake Walk%"cake"LOAD RUN  
%...  
%-----  
%  
MENU  
-----

The MENUDISP word displays the following lines starting with % as a menu screen. Normal % lines are simply printed on the screen. Those with a dividing % in the middle are option lines - the first part is the screen text, and the second is the command line for the option. The MENU word takes the complete display and runs it as a menu.

Option lines can be up to 35 characters long, excluding the % characters. The complete menu should not be more than 24 lines long,

to leave at least one line for the command area.

To call your menu, you just enter its name as a command. When you select an option, the corresponding command line is executed, leaving you in command mode. In the case of a jukebox menu, the menu program is replaced by the loaded program, but in other cases you can make the system return to the menu after doing an option by adding the menu word as the option's last command.

You are free to put menu words as options in other menus, so you can have menus leading to further menus. For example, if you have more options than will fit on one screen, divide them into separate menus connected in a ring, with each one calling the next as the first option. In a tree structure, the first option calls the previous menu, so that you only need to press RETURN to re-trace your steps.

## the !BOOT file

The system file !BOOT sets up the AMPLE system from both disc and cassette. You can add extra commands to the file, for example, a keyboard auto-repeat rate setting, using a text editor or word processor. They should go immediately after the \*BASIC line. If you want to add function key definitions of your own, these must go after the PREPARE line.

## keeping system files separately

If you have a double disc drive, you can keep module files and programs on separate discs, giving more disc space and files for programs. You do this by adding the specifier of the system drive to the filename prefix set by MPREFIX, and setting the current default drive to that of the program disc. There is already an "M."MPREFIX instruction in the !BOOT file so that modules are loaded from directory 'M'. You add the drive specifier before the 'M'.

For example, to use a system disc in drive 0 and a program disc in drive 1 on the DFS, use

```
" :O.M."MPREFIX      % :O.filename for module loads
*DRIVE 1             % other files, inc. program, on drive 1
```

The DRIVE command is added on the line after the MPREFIX.

The command AMPLE sets MPREFIX to "" (null).

## memory usage on standard computers

The !BOOT file command \*/C.PREPARE maximises the amount of memory available for programs. On standard DFS disc computers and other DFS computers that have no additional service ROMs, it reclaims 1.5K of memory from the DFS. The restrictions are that you cannot:

- \* change between disc and tape filing systems
- \* use SPOOL and EXEC at the same time, though you can individually

Language ROMs do not decrease user memory, so they don't prevent reclamation. Service ROMs that increase the total memory used by the operating system do prevent it. On standard ADFS disc computers, more memory is available to the system anyway, and PREPARE has no effect on the memory usage.

Note: PREPARE has the effect of setting OSHWM to &1300 if, and only if, it was &1900 originally. Further restrictions are that you cannot have more than one sequential file open at a time, or use additional service that use non-maskable interrupts (NMIs).

## memory usage on non-standard computers

You can still use the system if your computer is non-standard, but with some restrictions. If your computer has a non-standard DFS and/or ROMs which decrease the amount of user memory, PREPARE cannot reclaim memory and you will have less memory available for programs. If this is a problem but the computer does have a standard DFS, it may be worth removing those ROMs that consume memory.

In the unlikely event that your non-standard computer has OSHWM at &1900, you must manually disable reclamation to avoid problems - removing the 'R' after PREPARE in the !BOOT file, so the line reads:

```
*/C.PREPARE C
```

## shadow RAM

AMPLE is compatible with built-in shadow RAM, and add-on shadow RAM that supports the official OSBYTE 111 shadow RAM control. Shadow RAM allows you bigger programs when using the Staff Editor.

If your computer has built-in shadow RAM, or add-on shadow RAM that is engaged by default, then AMPLE will use it automatically. (If the computer is not a Model B, the PREPARE command issues a \*SHADOW command to engage shadow RAM).

If your computer has add-on shadow RAM that is not selected by

## Further use of AMPLE

default, you must select it before entering AMPLE. If there is a simple command available to engage the shadow RAM, you can add it to the !BOOT file, right after the \*BASIC line.

Note: Without shadow RAM, the maximum size of program you have with any editor is limited by the Staff Editor because it uses a graphics screen. Shadow RAM reclaims the screen memory (8K) so that the largest program you could have with the Mixing Desk or Notepad can also fit in with the Staff Editor. Shadow RAM makes little difference when the Staff Editor is not in use.

## ROM compatibility

AMPLE makes use of its language memory through interrupts at all times including during operating system functions, using entirely correct and proper means. For this reason, commands that 'borrow' language memory, for example DFS \*COPY, cannot be used from within AMPLE.

Some ROMs also 'borrow' language memory on certain operations, usually '\*' command or \*HELP command processing. The result is that these ROMs corrupt AMPLE workspace when '\*' commands are issued from AMPLE, possibly crashing the system. Certain word processors and 'toolkit' ROMs are known to do this. If you find a problem, successively remove additional ROMs until it is cured.

## memory map

The map of AMPLE's memory usage is as follows:

&8000	-----	top of screen
	screen	
	-----	top of language memory (&8000 for shadow RAM)
	user program	
	and modules	
OSHWM	-----	bottom of language memory
	OS workspace	
&0B00	-----	
	Staff buffer	(OS general workspace)
&0900	-----	
	OS workspace	
&0180	-----	The memory from &900 to &AFF can be used for general OS functions when the Staff Editor data (the music on the staff) is not needed.
	AMPLE w'space	
&0100	-----	
	OS workspace	
&0080	-----	
	AMPLE w'space	
&0000	-----	

## Reference section



## 10 AMPLE commands

This chapter gives concise descriptions of the most important AMPLE commands, and others that are not in fact commands, but are for use inside words.

**[ ... ] create word**  
namestring [ ... ]

You can create a word without using an editor by entering its name and definition directly at the keyboard, using [ ... ] to enclose the contents. The definition can stretch over more than one line, and the normal % prompt is replaced by [% to show when you are inside [ ... ].

```
example      "slap" [ 12 2 Echo ]
              "cres" [ 40=L
                      20 16 +L ]
```

### AMPLE restart installation

The command AMPLE completely restarts the installation, preserving only fixed modules. It checks that the ROM image is correct and sets MPREFIX to "" (null). Following commands on the same line are ignored. You can use this command to restart the system without reloading it. Use it with care since it discards the program.

```
example      AMPLE          % restart the system
              "M."MPREFIX"  % then restore the module prefix
```

### CLEAR clear editor data

CLEAR clears the data in the editor, leaving it ready for new data. The Staff Editor preserves its data between uses, so you should use CLEAR when starting afresh with it. In the Mixing Desk, CLEAR also sets-up the voices previously specified with SHARE and VOICES.

### **COMPACT compact unused memory**

COMPACT makes sure that all unused memory is available for use. If you get 'No room' on an important operation, do COMPACT and try again.

### **DELETE delete user word**

namestring DELETE

You use DELETE to remove single user words that are no longer required. You cannot delete a word that is in use by another word.

**example**            "part1"DELETE    % delete the word 'part1'

### **DISPLAY display text lines**

DISPLAY displays the following lines starting with %. It is used to make screen displays, particularly RUN title displays.

**example**            See RUN.

### **FIND find uses of word**

namestring FIND

FIND displays the names of all program and preset words which use the named word. You can also use it on 'INS1' to find all uses of preset instruments.

**examples**            "CHANS"FIND        % list names of all instruments  
                      "INS1"FIND        % find all uses of INS1 instruments

### **GET get word into editor**

namestring GET

GET gets the named word into the current editor so it can be examined or edited. Non-readable words, such as normal AMPLE system words, appear empty.

**example**            "part1"GET        % get 'part1' for editing



**LOAD load program**

namestring LOAD

LOAD does NEW and then loads the named file as a user program. The contents of Notepad or the Staff Editor are not affected. LOAD has the same function as the 'Load program' option on the Main Menu.

**example**            "tunes"LOAD        % load 'tunes'

**MAIN call-up main menu**

MAIN calls-up the Main Menu, exiting the current editor.

**MAKE make word from editor**

MAKE makes a word from the current contents of the editor, giving it the current name, as displayed at the top of the editor screen. The new word replaces any previous word of the same name.

**MEM show memory usage**

MEM shows the amount of user memory in bytes used for the following:

Words	user words (the program)
Data	for special editors - not used by Studio 5000 editors
System	players and special effects like Len and Echo
Arrays	store requested by the program (advanced programs only)
Free	the amount of memory remaining unused

The sum of these is the total amount of user memory. This depends on the operating system memory usage, the screen mode, and the modules loaded. Some free memory may be unavailable for use - see COMPACT.

**MENU present menu**

MENU presents the menu display printed up by MENU DISP, allowing the user to make a selection by moving the cursor and pressing RETURN. It is used at the end of a user menu word, with MENU DISP at the start.

## AMPLE commands

```
example      "cmenu" [ MENUISP % 'cmenu' to enter
%              AMPLE Commands
%-----
% Show user words%SHOW cmenu
% List program%WRITE cmenu
% Show memory usage%MEM cmenu
% List instruments%"CHANS"FIND cmenu
%-----
MENU ]
```

## MENUISP display menu

MENUISP displays the following lines starting with % as a menu screen.

A following line is either a simple display line, or if it has a further, dividing % is an option line, with the first part being the displayed text and the second the command. The displayed text and command together can be up to 35 lines long. The total number of lines should be less than 25. The last line is followed by MENU, which presents the menu.

**example** See MENU

## NAME set name of word in editor

namestring NAME

NAME sets the name to be used for the new word by MAKE. When you use GET on a user word, the 'making' name is set automatically. The name must be less than fifteen characters and use no upper case. Avoid names that use note letters only, for example, 'abc' and 'face'.

**example** "part2a"NAME MAKE % set name to 'part2a', and make

## NEW discard program

NEW discards all the user words and any modules automatically loaded by the program, leaving you a blank program from which to start a new one. Use it with care - the old program cannot be recovered.

NEW is equivalent to the 'New program' menu option.

**PLAY play sections and players with mixes**

liststring PLAY

PLAY plays a piece from its user-word elements. The string has a list of identifiers which say which score and mix words are to play, the order in which they play, and which players play.

For a simple piece with the score words 'part1', 'part2' etc and a word 'mix', you just put a list of player numbers:

```
"1234" PLAY           % players 1 to 4 for a four-part piece
```

If you have section score words like 'part1a', 'part1b' for each player, you add a list of section letters at the end, plus a '-':

```
"1234-ababc" PLAY      % play sections in order ababc
```

If you have different mixes called 'mix1', 'mix2' etc, you include mix numbers to call them up between sections:

```
"1234-1ab2ab3c" PLAY
```

If you don't put a mix number at the start, 'mix' is used instead.

```
examples      "123-abac" PLAY           % four sections on three players
               "123-1a2a3a" PLAY        % same section with three mixes
               "23-abac" PLAY           % players 2 and 3 only
```

**QUIT leave editor**

QUIT leaves the current editor, making the full screen available as a command area. You don't need to QUIT before switching to another editor. To return to the editor, select it from the menu.

**READY ready system**

READY readies the system to start a session of editing or playing. It does the following:

- \* stops the program, freeing memory used by players
- \* silences and frees all voices, so they can be used at keyboard
- \* sets the tempo to normal: 48, 125=T (125 crotchets per minute)
- \* resets all music values by doing a SCORE
- \* returns SHARE to use the keyboard's voices (0 SHARE)

READY is equivalent to the 'Ready system' menu option.

## **RENAME**   **rename word**

oldstring newstring RENAME

RENAME changes the name of an existing user word, including all its uses in the program. It does not change the Editor contents. If you end up with two words of the same name, a warning is given, and you should then rename again to a unique name.

**example**   "part1" "part1a" RENAME   % rename 'part1' to 'part1a'

## **RUN**   **run program**

RUN is not a system word but a user word that runs the program. All complete programs should have a RUN word, usually containing:

- \* DISPLAY to display the following comment lines
- \* Title text on comment lines (starting with %)
- \* a PLAY instruction

It is a good idea to keep the title display below 14 lines so that it fits in the command areas of the Mixing Desk and most menus.

If you enter RUN when there is no program, you will get 'Mistake'.

The command RUN is equivalent to 'Run program' on the Main Menu.

**example**            in Notepad:

-----  
DISPLAY

%

%                    'The Blue Danube'

%                                by

%                                Johann Strauss

%

"12-1aabbcc" PLAY

-----

## **SAVE**   **save program**

namestring SAVE

SAVE saves the program in memory, that is, all user words, as a file for reloading with LOAD. The Editor data is not saved.

SAVE has the same function as 'Save program' on the Main Menu.

**example**            "prog2"SAVE        % save program as file called 'prog2'

## **SHARE use specified player's voices**

number SHARE

SHARE controls to which player sound instructions will be sent. You use it to send VOICES settings to players when setting-up a mix, and to select a player's voices when adjusting an instrument.

READY and the 'Ready system' menu option perform a '0 SHARE', selecting the voices of the computer keyboard, rather than those of any player. You should start each session of playing tunes from either Notepad or the Staff Editor with a READY to make sure that you are using the keyboards voices.

## **SHOW show user words**

SHOW displays the names of all user words in alphabetical order, and the number saying how many there are. You use it to find out what words you have in the program at any time.

Note: SHOW also prints the message 'no data': this refers to a feature of future advanced editors, and should be ignored.

## **TYPE type contents of named word**

namestring TYPE

TYPE displays the contents (definition) of the named word on the screen, in text form. This lets you examine a word without getting it into an Editor, and is most useful for text-form words, rather than, for example, mix words. Non-readable words, such as normal AMPLE system words, appear empty.

## **VOICE select player's voice**

number VOICE

VOICE is used to select one or all of the voices of the player selected with SHARE. You can use it to put different instruments on individual voices directly, without the Mixing Desk. EVERY VOICE selects all voices, and this is done automatically by SHARE.

**example**            3 VOICES            % three voices  
                      1 VOICE Drum        % with different instruments  
                      2 VOICE Cymbal  
                      3 VOICE Yakbell

**VOICES set number of voices**

number VOICES

VOICES sets the number of voices to be used by the player. You use it with SHARE when setting up a mix, and when getting voices at the keyboard to try instruments and tunes on. Note that if you are only asking for one voice you still use VOICES - VOICE does something different.

**examples**           READY 3 VOICES Moog    % three voices at keyboard  
                  1 SHARE 1 VOICES       % one voice for player 1 in a mix

**WRITE display contents of all words**

WRITE displays the contents of all words on the screen in text form. You use it to produce program listings, and text files for adding to other programs and editing with a word processor.

WRITE arranges the output with empty definitions of the words first, so that the real definitions can be re-entered in any order.

**examples**           CTRL B WRITE CTRL C       % list program to printer  
                  \*SPPOOL prgtext           % spool complete text of  
                  WRITE                    % program to a file  
                  \*SPPOOL

## 11 AMPLE music words

This section contains detailed reference-form descriptions of the AMPLE music notation words, with a separate entry for each one. The entries are in the following dictionary-type order:

! ' ( ) + , - / : ; = @ \ ^ ^ ; | ~  
A-G a-g BAR K( )K 'L =L +L -L SCORE =T +T -T X

The form of each entry is as follows:

word name      function  
form of use (showing any numbers required)  
  
description  
  
example(s)  
  
related words  
  
further information (for advanced users)

### !    move an extra octave up or down

The '!' ('pling') word moves the pitch of the next note up (if upper case) or down (if lower case) by one octave, and hence raises or lowers all the following notes. You use it to make pitch jumps of more than an octave, and jumps of an octave between notes of the same case.

**examples**            0:c !C    is equivalent to    0:c 2:c  
                      0:c !c    is equivalent to    0:c -1:c  
                      0:C !C    is equivalent to    0:C 1:C

**related words**    :

### '    accent next note or hit

' is the accent symbol. It goes before the note or hit you want to accent, and increases the dynamic level of that note only.

The dynamic level is normally at the maximum of 64, so unless you lower it first, you will not hear the effect of the accents. 49 is the highest level that allows the full effect of accents - you can use

## Music words

a higher level, but the accent effect will be weaker.

You accent a whole chord by putting ' before the main note. You cannot put ' inside the brackets unless the chord is a broken chord, in which case each note (including the main note) may be accented separately.

The amount of dynamic level that the accent adds to the note is normally 15, but this can be set to other values using 'L.

' stops any +L or -L in progress at the current value.

<b>examples</b>	lower main level for accents	SCORE 49=L
	percussion	'X/// //X'X /X/'X
	melody	'CaBC 'aB'D//cb^
	chord	'G(BDF)
	broken chord	'G(12,'B'D'F)
	broken chord, top note only	G(12,BD'F)

**related words** 'L=L

## ( start additional chord notes

Chords are written using round brackets. A chord consists of a first ('main') note, followed by the other notes enclosed in round brackets. In other words, the brackets contain those notes that are to play at the same time as the previous note.



The pitch of the main notes in a chord sequence moves in the normal way (upper-case up, lower-case down) from main note to main note. The pitch of the bracketed notes starts from the main note, and then moves from note to note inside the brackets (the bracketed notes do not affect main notes), for example:

in the sequence:	c(EG)	D(FA)	c(EG)	D(FA)
the main note plays:	c	D	c	D

The bracketed notes can go up, down, or up and down from the main pitch.



Rests and holds can be included in the main line or in the brackets to stop the sound of notes on individual voices. You do this to play a chord which has fewer notes than the one before it, for example:

```
C(EGB) F(AC^ ) F(AC)
```

A full chord of rests (for example ^(^) ) stops all voices, playing a rest for the whole part. A more convenient way of doing this is with the chord rest symbol ^; (this acts on all voices put on the player by the last mix word).

If no symbol appears for a particular voice, it continues as if a hold had been written. This means that a hold on the main voice will hold the complete chord, so you never need to write a full chord of holds. For example, / is equivalent to /(///) .

You can play broken chords (chords with a strumming effect produced by a delay before each additional note in the chord) by including a length setting after the open bracket. Each note after the setting will be given a delay of that amount before it. This length setting only applies inside the brackets - it does not affect the length setting of the main notes. Also, the delays don't affect the main note's time, so they don't make the whole chord longer.

This shows the effect of a length setting inside brackets:

	normal chords	broken chords
	48, C(EGB)F(ACE) ^;	48, C(8,EGB)F(8,ACE) ^;
Voice 4	B-----E-----^	B-----E--^
Voice 3	G-----C-----^	G-----C---^
Voice 2	E-----A-----^	E-----A----^
Voice 1	C-----F-----^	C-----F-----^

To make broken chords stop completely before the next strum, you can 'dampen' all voices by putting a zero-length chord rest between them, for example

```
48,C(8,EGB) 0,^; 48,F(8,ACE) ^;
```

Hits (X s) can be included in chords to restrike previous notes or to play percussion instruments. Even if you are using symbols that play in particular voices (your own definitions, for example), you can still use brackets to play two or more hits together.

Remember that to play chords, you must put an instrument on each of the voices you want to play. This is normally done with the Mixing Desk.

## Music words

### examples

simple sequence	c(EG)D(FA)c(EG)D(FA)
isolated chord	C(EG-B) ^;
elaborate sequence	C(EG) /( /a) g(BD) / f(A^)
broken chords	C(8,EGB)F(8,ACE)
moving main voice	C(F)bag f(D)GAB C(F)/// ^;
percussion with X	X/ /(X)/ X/ X(X)
percussion with user symbols	"x" [1;X] "y" [2;X] ... x///y//x x///y//x(y) x

**related words** ) ; ^; ,

### further information

When scoring broken chords, the total delay inside the brackets can even be longer than the main note, making the strummings overlap. You can also set a negative delay so that the strum goes before the main note, leading up to the beat.

	overlapping	leading
	48,C(24,EGB)F(24,ACE)	C(-8,EGB)F(-8,ACE)^;
Voice 4	B-----E-----	B-----E-----^
Voice 3	G-----C-----	G-----C-----^
Voice 2	E-----A-----	E-----A-----^
Voice 1	C-----F-----	C-----F-----^

When using overlapping and leading chords, keep in mind that you are actually scoring notes to play in the future and the past, and watch out for them reaching past other events in the present. For example, if you were to add a chord rest (^;) to the end of the overlapping example above, it would silence the F, A and C, and the B from the previous chord. The final E would then play as normal. A broken chord of rests, ^(24,^^) would silence all the notes.

SCORE does a '1;' so unless you use ';' yourself, normal notes will always play on voice 1, and additional chord notes will start on voice 2. You can use ';' outside the brackets to make the chord start on a higher voice.

The dynamic level and accent normally apply to all notes in a chord, but settings can be put in the brackets to affect individual notes.

You cannot put more chord brackets or a key signature inside chord brackets.

'(' leaves some numbers for use by ')'. These values are not defined, and you must not interfere with them.

## ) end additional chord notes

Chords are written using round brackets. See ( for more information.

## + sharpen next note

+ is the sharp sign. It raises the pitch of the next note by one semitone, overriding the key signature. It affects the next note only.

<b>examples</b>	+F	% F sharp
	++C	% C double-sharp

**related words** - = K(

## further information

The total modification of the pitch can be up to plus or minus 64 semitones.

## , set length

number,

',' sets the length for notes, hits, rests, and holds. You use it at the start of each phrase to set a suitable basic length, and longer notes are made by extending from this using holds. If a short note or irregular length (like triplet) appears in the middle of a phrase, you often make a temporary change of length. The number is usually in the range 0 to 192.

Here are suggested length values for the common note values:

hemidemisemiquaver	3	demisemiquaver	6
semiquaver	12	quaver	24
crotchet	48	minim	96
semibreve	192	breve	384

## Music words

To get the length for irregular note values such as triplet and duplet, just multiply the normal length by the appropriate factor:

modified value	factor	example
triplet note	2/3	quaver: 24 -> 16
duplet note	3/2	crotchet: 48 -> 72

You can do the same for dotted notes, but these are better written as three half-lengths, using holds.

You can set short lengths for grace notes in ornaments such as trills and mordents. A basic length of zero is sometimes useful, for example, for a rest which ends a note but takes up no time itself.

The length can be set inside chord brackets for arpeggiation effects. See '(' for more information.

Every one of your score words (words containing music words) should start with a ',' setting, unless you deliberately want it to use an external setting.

<b>examples</b>	set crotchets	48,
	set at start of phrase	24,C/// fGAB C//B C/// ^
	change in middle of phrase	16,GGG 24,G/G/GGG/

**related words** / \ BAR |

### further information

Which length you choose to represent a particular note value such as a crotchet is entirely up to you, since the tempo can be adjusted over a wide range to give the correct playing speed. Large values are cumbersome and small values cannot be as far divided for short notes - the suggested values are a good compromise.

The full range of the length is -32768 to 32767. The ability to program negative-length events is a very advanced feature which can be used for articulation effects, backwards-playing notes, overlaid note sequences, random access to the time domain, and more. See '\\' (back-hold) for more information.

### - flatten next note or indicate negative number

- is the flat sign. It lowers the pitch of the next note by one semitone, overriding the key signature. It affects the next note only.

The '-' character acts as a minus sign before a decimal number.

**examples**

-B	% B flat
--E	% E double-flat
-200	% minus 200

**related words** + = K(

### further information

The total modification of the pitch can be up to plus or minus 64 semitones.

## / hold notes, hits, and rests

'/' is the hold symbol. It holds all voices for one basic length. It is used to extend notes, hits, and rests as an alternative to changing the basic length.

The hold is the most basic AMPLE music symbol since it simply marks a unit of musical time in which notes, hits, and rests all continue with no change. It is often used where a rest would appear in more specialised music notations, for example, between percussion hits, and in long sections where a part doesn't play.

'/' affects chords exactly like normal notes, so single '/' in the main voice holds all the voices. There is no point in writing a full chord of holds, since the (///) is redundant. You do use '/' in brackets to hold voices when a higher voice is not being held.

### examples

tune with simple rhythm	C/// fGAB C//a g///
two crotchets tied across bar line	48,f   /
'rests' with percussion instrument	X/// /X/X
four bars 'rest'	192,^   / / / /   or
	0,^ 192, / / / / /
held chord	C(GE) / C(AF) C(BG)
passing note in chord sequence	C(EA) /( /G) a(DG)

**related words** , \

### further information

The length of the hold is added to the bar's total of note lengths for checking by the next bar line.

0,/ has no effect on the music. This is normally the case in chord brackets, but the hold still passes over its voice so a non-hold can

## Music words

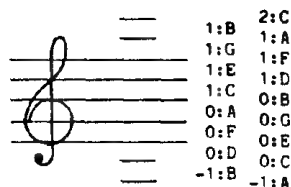
be put on a higher voice.

### : set octave

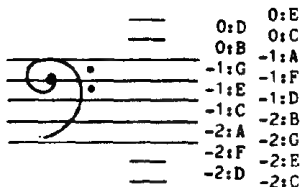
number:

':' fixes the pitch of the next note. The numbers indicate the octave, with 0 being the middle of the pitch range, so for example, 0:C is middle C, 1:C is the C above and -1:C is the C below. 0:D , 0:E , 0:F etc. are the pitches immediately above middle C. AMPLE pitches for the lines and spaces of the clefs are as follows:

treble clef



bass clef



A lower-case note goes down from the ':' setting, so 0:b , 0:a , 0:g etc. are just below middle C; the same as -1:B , -1:A , -1:G etc.

middle C

0:c 0:d 0:e 0:f 0:g 0:a 0:b 0:C 0:D 0:E 0:F 0:G 0:A 0:B  
<-----lower octave-----> <-----upper octave----->

**example**      0:C/// fGAB C//a g///    % play tune starting at middle C

**related words**    !

### further information

':' actually sets the effective last note pitch to be C in the numbered octave, so for example 0:C sets the pitch as if middle C had just played from an upper-case C.

### ; set music voice

number;

';' sets which voice notes, hits and rests are played on. For chords, you use chord brackets which select the voice automatically, but in other, special, cases you use ';' for individual notes on different voices. For example, for

\* setting the voice in a user-defined percussion word which always hits a particular voice

- \* scoring sequences of overlapping notes
- \* playing on all voices simultaneously

Since the length of a note, rest or hit affects all voices on the player (like a hold), the voices always stay in step. Each voice continues playing its note, rest or hit until another one on that voice, unaffected by what happens on other voices meanwhile. Holds affect all voices, whatever the voice setting.

SCORE does a '1;' so unless you use ';' yourself, normal notes will always play on voice 1, and additional chord notes will start on voice 2.

The upper/lower case pitch movement is common to all voices on a player, so the pitch moves from note to note as normal, whatever the voice.

Remember that you must put an instrument on each of the voices you want to play.

<b>examples</b>	use voice 1	1;
	chord build-up effect	16,1;cDEF G 2;fGA B 3;aBCD
	overlapping notes	1;C 2;G 1;D 2;A 1;E 2;B
	percussion hit symbols	"x" [1;X] "y" [2;X]
	and used in	...
	percussion score	x///y//x x///y//x(y)

**related words** (

**further information**

EVERY; (or ON;) selects all the player's voices simultaneously. This is useful for, for example, restriking all notes of a chord with X :

repeated chord	A(CE) A(CE) A(CE) A(CE) A(CE) A(CE) //
alternative form	A(CE) EVERY;XXXXX// 1;

**= naturalise next note**

= is the natural sign. It cancels the effect of the key signature on the next note, so it plays at its unmodified pitch. It affects the next note only.

<b>examples</b>	=b	% B natural
	=F	% F natural

**related words** + - K( )K

## @ set transposition in semitones

number @

@ sets the transposition to be applied to notes. It is used to automatically transpose the pitch of sections, without changing the letter names etc. The number before it is the transposition in semitones, which can be positive or negative.

@ has a variety of uses, including:

- \* scoring music for transposing instruments
- \* playing a fixed note pattern ('riff') at different pitches
- \* moving a part to play at an octave above or below.

The following table shows the @ value (number of semitones) for transposing from C:

transpose C to lower pitch:

Pitch name	C	C#	D	D#	E	F	F#	G	G#	A	A#	B	C
AMPLE name	c	+c	d	+d	e	f	+f	g	+g	a	+a	b	C
@ value	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0

transpose C to higher pitch:

Pitch name	C	C#	D	D#	E	F	F#	G	G#	A	A#	B	C
AMPLE name	C	+C	D	+D	E	F	+F	G	+G	A	+A	B	!C
@ value	0	1	2	3	4	5	6	7	8	9	10	11	12

SCORE cancels the transposition, that is, it does a 0@.

The key signature works on the note letters rather than the note pitches, so you don't need to worry about the effect of transposition on it.

<b>examples</b>	play music an octave up	SCORE 12@ ...
	play at root, fifth, fourth	0@ riff 7@ riff 5@ riff
	play music for a Bb clarinet	SCORE -2@ ...

## \ back hold

\ is the converse of / - it moves back in music time by the basic length setting. This back-spacing effectively undoes the net time effect of previous holds, notes, rests and ties, but doesn't affect when they play, only affecting when the following notes play. It is useful when you find it more convenient to score notes in an order other than that in which they play, for example, to overlay two separately scored note sequences.



The back hold symbol lets you reach from one point in the score, say, where you have used a music symbol of your own definition, to place a note at another point in the past or future. One example of this is when a part is broken up into sections and defined as separate words. If a verse was to start with a few notes which actually played before the start of the first bar, that is in the last bar of the previous section (a pick-up), you could score it at the start of the verse where it belongs by first back-spacing with the back hold symbol.

When moving backwards and forwards in time with \ and / , watch out for spacing past other sound events like instrument changes.

In all other respects, including the effect inside chords, \ is exactly like / . In fact,  $48, \backslash$  is equivalent to  $-48, /$  , and  $-48, \backslash$  is equivalent to  $48, /$  .

## examples

```
do nothing                                /\  or  \/

three-note pickup                        12,\\ \\ DEF G/// /fed c///

percussion score ...                     24,X/// X/// X///
with flam (double hit) ...               24,X/// X//20,/ 4, X 24,X///
then written with \                      24,X/// X/// 4,\X 24,X///

% interleave notes of two riffs, half a beat apart
riff1 48,\\ \\ \\ \\ \\ \\ \\ \\ \\ \\ % back to start
24,/ riff2 24,\ % forward half beat, then back
```

**related words** / , ( )

**further information**

There is no limit to the amount of time by which you can back-space with \ , but there is a limit to the number of sound events you back-space past. This limit is equivalent to about 50 notes minimum, that is, including those on other players. You will not normally reach this limit.

The system does not play past the current position of time, so for example if you are entering notes at the keyboard and back-space past some notes with \ , they will not be played until you forward-space to a point after them with / or notes, hits or rests.

## **^ play rest**

**^** is the symbol for a rest. It plays a rest lasting for the basic length, on the current music voice. It finishes the previous note on that voice, causing the sound to stop or die away depending on the instrument in use.

Rests can be extended in length by the hold symbol. This is preferable to repeating the rest, for example, **^///** is better than **^^^^**. Hold symbols should also be used rather than rests to mark passages where a part doesn't play. In percussion scores, normal non-hit beats ('rests' in other notations) are marked by hold symbols: the rest has the effect of cutting short the sound of the previous hit, just as it ends the sound of a note.

Since rests work on individual voices, to silence all notes of a chord, you should use **^;**

Each part of a piece usually ends with rests to finish the last note(s). Section ends can also include rests so that the notes always finish whatever is notated at the beginning of the next section. In this case, they must be zero-length so they take up no musical time.

<b>examples</b>	<b>C^C^C^C^</b>	<b>% isolated notes</b>
	<b>192,D 1 ^</b>	<b>% end of piece</b>
	<b>X///X^/X X///X^X^</b>	<b>% decaying and cut-short beats</b>
	<b>CEGC 0,^</b>	<b>% zero-length rest to silence</b>

### **further information**

The length of the rest is added to the bar's total of note lengths for checking by the next bar line.

**related words** **^;**

## **^; play chord rest**

**^;** plays a chord rest (a rest on all voices of the part), lasting for the length setting. It is equivalent to **^(^...^)** with a rest symbol for each of the voices, but more convenient for use with chords.

**example** **C(EG-BC)^;C(EG-BC)^;**

**related words** **^ ;**

### **further information**

**^;** is equivalent to **EVERY;^**, but it does not change the **;** setting. It has the effect of **^** on each of the voices in the range set with

VOICES.

## **| mark end of bar**

| represents a bar line. Bar lines are used only to check the length of bars (detecting extra or missing items) and have no effect on the music.

The use of bar lines is entirely optional. For full scores translated from written music with bar lines, and music with many basic length changes, they are usually worth using. For simple tunes and music composed directly into AMPLE, they are usually best left out.

The first bar starts with the SCORE, so there should be no bar line before it. Bar lines are placed at regular intervals throughout the part, the last one going at the end but before any extra rest you have added to finish the final note.

BAR sets the desired bar length, and if a bar line finds that the total of lengths since the last bar line or SCORE is different from this, then it gives the 'Bad bar' error.

If the bar length is set to zero, lengths are still totalled but checking is disabled. If you are trying out extracts at the keyboard by cursor-copying incomplete bars, you should set 0 BAR to avoid unwanted bar errors.

SCORE sets the bar length to zero, so you must use BAR if bar checking is required.

<b>examples</b>	...   48, 4 BAR 48,C E F c	% ok
	...   48, 4 BAR 48,C E D	% gives 'Bad bar'
	...   0 BAR 48,C E D	% not faulted

**related words** BAR

## **~ slur next note**

~ causes the following note to be slurred, that is, played as a continuation of the previous note without re-starting the envelopes, and without any gap set by 'Len'. You put a slur between two notes that you want to be played in a smooth connected fashion, with just the change of pitch between them.

The effect of a slur depends strongly on the envelope. On envelopes with a non-zero sustain level, it sounds as you would expect, but on those that decay to zero, the second note is quieter and may be lost entirely. Note that 'slurs' on piano-type sounds are in fact gate

## Music words

period effects and are not produced with the slur symbol.

Individual notes of a chord may be slurred. There is no point in slurring hits, rests or notes with repeated pitch, since the result is the same as a hold in each case.

<b>examples</b>	melody	Dc b~ggd~GG B~gBD~cb
	chords	A(CE) /( /~F) A(DF) ~C(~E~G) ^;

## A to G play note with ascending pitch

The letters 'A' to 'G' play notes of their respective pitches above the previous pitch.

AMPLE music notation uses the letters A-G to represent the seven main note pitches (the white keys on a piano keyboard). In upper-case (A-G), the letter plays that pitch above the last pitch, and in lower-case (a-g), it plays the pitch below. The exception is that a repeated letter (with the same case) always plays the same pitch.

This scheme lets you write out phrases of music as a sequence of note letters with no octave indications: each note's pitch is up or down from the previous one, depending on its case. The octave pitch of the whole phrase is set with a single ':' at the start. For intervals greater than an octave, you use '!' to cause an extra octave jump on the following note.

The note takes the length set with ',', and you can extend it with the hold symbol, /.

The '+' (sharp) and '-' (flat) symbols modify the pitch of the following note by one semitone. A key signature ( K(...)K containing a list of sharpened or flattened notes) modifies all uses of particular note letters, except those that have a '+', '-' or '=' symbol (accidental) on them. @ transposes all note pitches by a specified number of semitones.

Notes normally play on the current music voice (set by ; ), but inside chord brackets ( (...) ), they play on successive voices starting on the current music voice. Chord brackets also set the length to zero temporarily, so the notes start at the same time.

<b>examples</b>	rising scale	CDEFGAB^
	falling scale	Cbagfedc^
	repeated note	CCCC^
	alternating octaves	cCcCcCcC^
	tune	EEFG gfed ccDE Edd^

**related words** ! : + - = K( @

### **further information**

The length of the note is added to the bar's total of note lengths for checking by the next bar line.

### **a to g play note with descending pitch**

The letters 'a' to 'g' play notes of their respective pitches below the previous pitch.

See 'A to G' for more information.

### **BAR set bar length in length units**

number BAR

BAR sets the bar length in beats (',' lengths) for checking by bar lines. You usually use it with a length setting beforehand to indicate the length of the beat. (The length setting has the function of the bottom number of a staff time signature, and the BAR number is the top number).

With the bar length set to 0, the checking action of bar lines is disabled. SCORE does a 0 BAR, so you must use BAR if you want the bar lines to do the checking.

See | for more information.

<b>examples</b>	four crotchets per bar (4/4 time)	48, 4 BAR
	three quavers per bar (3/8 time)	24, 3 BAR

**related words** |

### **further information**

You can change the BAR setting at any point in music, for example, to disable bar checking for a deliberately-long bar, or accommodate a change in time signature.

**K( start key signature**

K( ... )K sets the key signature for the player. A key signature is a set of sharps and flats for particular note letters which are automatically applied to each of those note letters in future.

To set the key signature, enclose a list of the note letters (upper or lower case), each with a sharp or flat before it, inside K( ... )K:

$$K( \begin{array}{c} +F \\ \text{pitch modification} \end{array} \begin{array}{c} +C \\ \text{note letter} \end{array} +G ) K$$
 start key signature                      end key signature

The key signature is often put in a 'sig' word, along with SCORE and BAR, which is then called up by all sections.

An individual note can be returned to its unmodified pitch by putting a natural symbol, '=', before it. This effect applies to that note only. The sharp and flat symbols (+ - or =) also cancel the effect of the key signature for the next note.

Chord brackets and more key signatures are not allowed inside key signatures.

<b>examples</b>	C major	$K( \quad )K$
	A major	$K( \quad +F \quad +C \quad +G \quad )K$
	F major	$K( \quad -B \quad )K$

related words + - =

### further information

The key signature can be freely changed in the middle of music.

There is no restriction on the modifications inside key signatures, so non-standard key signatures can be created. This is particularly useful for minor keys, since the raised 7th can be included, for example, D minor: K( -B +C )K.

Notes inside key signatures do not play or alter the effective previous note pitch.

)K end key signature

$K(\dots)$  sets the key signature for the player.

See K1 for more information.

**'L set accent level**

number 'L

'L sets the amount of dynamic level added by ' (accent), that is, the strength of the accent. SCORE sets the accent level to 15 and this will be used until you change it with 'L. Suggested settings are:

```

10'L    weak
15'L    medium
20'L    strong
25'L    very strong
-15'L   negative (accented notes are quieter than normal ones)

```

Remember that the maximum dynamic level is 64. If the accent level plus the =L level is more than 64, you will not hear the full effect.

**example**            SCORE 30'L        % make accents stronger

**related words**    ' =L

**=L set dynamic level**

number =L

=L sets the dynamic level of the player's voices in the range 0 (soft) to 64 (loud). 64 is the normal value, set by SCORE. Since the level starts off at maximum, if you want to make a particular passage louder than the others, you must make the others quieter. The =L control is independent of the Mixing Desk volume control.

=L cancels any +L or -L level change in progress.

The dynamic level takes effect at the start of notes only, so, for example, you cannot change it in the middle of a note.

```

examples            40 =L                % set medium loud level

                     24=L 40 16 +L        % set level before crescendo

                     "pp1" [ 15 =L ]     % define words for common levels
                     "p1"  [ 25 =L ]
                     "mp1" [ 35 =L ]
                     "mf1" [ 45 =L ]
                     "f1"  [ 55 =L ]
                     "ff1" [ 64 =L ]

```

**related words**    +L -L '

## **+L increase dynamic level**

changenumbers eventsnumber +L

+L increases the dynamic level of the player's voices by a specified amount over a specified period of time. It is used for making changes of level relative to the current setting, both instantly and automatically over time (crescendo).

The first number is the amount of change in the dynamic level, usually in the range 0 to 64. The second number is the number of events, of the length setting in use at the time of the +L, over which the change is to take place. These events can be notes, rests, hits and ties, and the length setting can be changed after the +L without affecting the length of the change period. A value of zero makes the change happen instantly. After the end of the change period, the level stays at the final value.

-L has the opposite effect to +L, that is, it decreases the level (diminuendo). You could specify a negative value for +L to get the same effect.

The change follows a slope starting at the +L and ending at the end of the last event. This means that the first note is unaffected (unless the change is instant), and the first note that plays at the final level is the one after the last event on the slope. This may sound odd, but is just the effect you want in practice.

Remember that the range of the dynamic level itself is 0 to 64. Also note that the dynamic level only takes effect at the start of a note, not within it (a crescendo within a note is a feature of the instrument's envelope).

### **examples**

cresc. of 20 over four crotchets      48, 4 20 +L

drum roll with 8 beats crescendo      12, 40 8 +L      XXXX XXXX XXXX X  
alternative      48, 40 2 +L 12, XXXX XXXX XXXX X

dynamic level of 24, for a bar,      24 =L    CDEG  
then rising to 64 over a bar,      40 4 +L GABC  
then falling back to 24 over a bar      40 4 -L DcBa g

crescendo on notes of broken chord      4, 40 4 +L    48,D(4,FACD)

**related words**    =L -L

**further information**



The change is halted by another +L, -L, =L or SCORE instruction. You can use the instruction '0 0 +L' to halt a change and leave the level at that value.

+L cannot be used with a length setting of zero (in non-broken chords, for example). This gives the 'Divide by zero' error.'

### **-L decrease dynamic level**

changenumber eventsnumber -L

-L decreases the dynamic level by a specified amount over a specified period of time. It is used for making changes of level relative to the current setting, both instantly and automatically over time (diminuendo).

-L is exactly equivalent to +L, except that positive change values give a decrease (and negative change values give an increase). See +L.

### **SCORE reset music values**

SCORE resets all the music values to convenient initial settings, to start the player afresh for music words. You put SCORE at the start of each section of music to make sure it is not affected by the settings of a previous section.

When creating a score word (a word with music notation inside), you should always put a SCORE at the start, unless it is merely a continuation of a previous word and needs to use the current settings. (Each continuation word should still have ',' and ':' at the very start unless you particularly want it to use external settings.)

SCORE provides the following 'default' settings which you can take for granted until you need to change them:

K( )K	% blank key signature (C major or A minor)
0 BAR	% no time signature; bars are not checked
64=L	% maximum dynamic level
15'L	% moderate accent strength
0@	% no transposition
1;	% using voice 1 upwards
SIMPLEACT	% no special effects like Len or Echo

You should always set ',' after the SCORE, before the first word in the music. If you are using note letters, you must also set ':' before the first one.

Note that SCORE cancels any musical effects like Perc, Len and Echo.

## Music words

Unlike other music values, tempo is an overall parameter common to all players, so SCORE does not affect it. Tempo settings are reset by READY, and automatically by PLAY at the start of a piece.

### further information

SCORE carries out the following:

```
SIMPLEACT K( )K 0 BAR 48, 1; 0@ 0: 64=L 15'L
```

It also cancels +, -, ! etc.

READY does a SCORE.

### =T set tempo

number =T

Sets the tempo in beats per minute. You usually make a length setting first to indicate the beat.

=T is commonly used to set a tempo when trying out single sections from an editor. You usually set the tempo of the final piece with the Mixing Desk (which then includes an =T in the 'mix' word), and use +T or -T to make changes of tempo inside the music.

=T cancels any +T or -T changes in progress. The 'normal' tempo value is '48, 125 =T' (125 crotchet beats per second). READY sets the tempo to this.

<b>examples</b>	100 crotchets per minute	48, 100 =T
	50000 ticks per minute	1, 50000 =T
	set the tempo to 60 beats per minute, for current ',' value	60 =T

**related words** , +T -T

### further information

The range of tempo allowed is 2 to 1365 crotchet (48,) beats per second. The range of the product of the '=T' and ',' values is 92 to 65535.

### +T increase tempo

changenumbers beatsnumber +T

+T increases the tempo by a specified amount over a specified period of time. It is used for making changes of tempo relative to the current setting, both instantly and automatically over time (accelerando).

The first number is the amount of change in the tempo, in the range -128 to 128. Here are some approximate example values:

-127	25% (one quarter)
-64	50% (half)
-30	70%
-10	90%
0	100% (no change)
10	111%
30	138%
64	200% (double)
127	400% (four times)

You can just as easily use -T for slowing down. -T is the opposite of +T, that is, positive numbers cause slowing down.

The changenumber values are proportional and complementary, so that

- \* the musical effect of a relative change is the same at all tempos
- \* corresponding positive and negative changes are complementary, that is, they always return you to the original tempo

The second number is the number of beats, of the current length, over which the change is to take place. You normally have a length setting first to set the beat. The tempo changes continuously over the following music (further length settings do not affect the change).

To make an instant change of tempo, you can use +T with a beat number of zero. This is often preferable to an =T setting in music because it changes relative to the =T setting, leaving you able to set the overall tempo with a single =T at the start.

If there are many tempo settings in a piece, you may find it easier to score them in a separate 'conductor' part for this purpose. You use holds to mark the passage of time between changes. This is especially useful for repeated sequences of tempo changes which are used over different musical sections.

<b>examples</b>	gradually increase in tempo	SCORE 125 =T 20 16 +T C//D E/c/ EFG/ A///
	instantly double the tempo	64 0 +T
	faster and slower returning to original tempo	30 8 +T ... 30 8 -T
	for use in conductor part	"rubato" [
	fall behind the beat	20 2 -T //
	catch up and go ahead	40 4 +T ///
	fall back to the beat	20 2 -T // ]

**related words** , =T -T

**further information**

The formula for the tempo change is:

$$\text{new tempo} = \text{old tempo} \times 2^{(\text{changenumber}/64)}$$

**-T decrease tempo**

changenumber beatsnumber -T

-T decreases the tempo by a specified amount over a specified period of time. It is used for making changes of tempo relative to the current setting, both instantly and automatically over time (ritardando).

See +T.

**X play hit**

X plays a 'hit' (an unpitched note) on the current music voice. X is very like the note letters A-G except that it does not set the pitch. It is used in percussion scores to play hits ('beats') where the instrument itself sets the pitch, and in normal scores to restrike the last note.

The hit takes as its length the basic length (set with ','), though you can extend it with the hold symbol, /. Many percussion instruments have a sound that dies away immediately, so this will only affect the amount of time to the next hit, not the length of the sound. Holds are used, rather than rests, to mark silent beats between hits, allowing the sound to die away naturally. A rest cuts a long sound short.

hit (sounding beat)      X / ^  
                              |    |  
                              rest (cuts short previous hit)  
                              hold (non-sounding beat)

Hits can be played in chords (exactly like notes) for multi-voice percussion scores. Alternatively, you can define a user hit word specific to each voice - "name" [ n;X ] where n is the voice number. You then only need to use chord brackets when two or more hits play on the same beat:

y///z//yy///z(y)///

hit on voice 1                      hit on voice 2                      hits on both voices together

The pitch of a percussion instrument is determined as part of its definition, with PITCH.

Used in normal (pitched) scores, X serves to repeat the last note on that voice. It is useful for repeating chords using EVERY; .

<b>examples</b>	rhythm with one cut-off hit	XXXX X//X X^// X/X/
	two-voice rhythm	X/// X//X X///X(X) //(X)/
	repeated chords	C(GE) EVERY;XXX 1;

**related words** ;

**further information**

The length of the hit is added to the bar's total of note lengths for checking by the next bar line.



## 12 Special effects

Special effects are pre-defined extensions like instruments, so their names are in lower case with an upper-case initial.

### **Autopan use =L panning**

flag Autopan

'ON Autopan' directs the musical level to control the stereo position, for automatic panning effects. The Mixing Desk monitors the position. 'OFF Autopan' returns to normal. Autopan cannot be used with Slide.

<b>example</b>	ON Autopan	% turn on Autopan
	3 =L	% set position to full right
	48, -7 8 -L	% pan left by 7 over 8 beats
	0:cDEG CEgc	% score
	OFF Autopan	% turn off Autopan

### **Slide use =L pitch offset**

flag Slide

'ON Slide' directs the musical level to a pitch offset, allowing continuous pitch slide effects including portamentos, glissandos and bends. The offset takes effect on every note, hit, rest or hold, so you can control when a pitch step takes place. For a continuous slide on a note, use a short length setting and a sequence of holds. The pitch offset is in quarter semitone units. 'OFF Slide' returns to normal. Slide cannot be used with Autopan.

<b>example</b>	ON Slide	% turn on Slide (sets offset to 0)
	148, 2 8 -L	% down by 12 semitones over 8 beats
	0:c/// ///	
	...	
	OFF Slide	% Turn off Slide

### **Echo use multiple echo**

delaynumber voicesnumber Echo

'Echo' produces a true echo effect by repeating the music played on voice 1 on successive voices after a delay. The first number is the delay time between echoes in timebase units. It can be positive for normal echo or negative for pre-echo. The second number is the

## Special effects

number of voices to be used. These voices must have instruments on them for the echoes to play. These should be specified by VOICES when setting up the mix.

```
examples      24 2 Echo          % slapback echo
               -1:48,c//C cgAB C/E/ cbag

               72 4 Echo          % 'tape loop' echo
               0:48,CD/G ec//
               CgeC 24,BDg/////
```

## Len set length/gap of gate period

number Len

Len is used to modify the gate period (the sounding portion) of notes by either decreasing it by a fixed amount to produce a normal detached style of playing, or setting it to a fixed length to give a staccato style. If the number is negative, then a gap of that length (in timebase units) is inserted before each note. If the number is positive, then all notes play for that length of time only, and a variable-length gap makes up the time until the next note.

A small negative Len setting, such as '-4 Len', is very important for separating successive notes of an instrument that has no emphasis at the start of notes, such as some organ sounds.

```
examples      -4 Len            % Shorten the notes by -4, to make each
               4B,cDEF G///      % note slightly detached from one another
               Cged c/c/
               +6 Len            % Make the note length 6, even though the
               24,CeGeCeFG       % beat length is 24, to play staccato.
```

## Perc use pitches as hits

'Perc' reconfigures the letter notes so that they play hits on different voices, allowing multi-part percussion to be scored on a single staff or single player.

letter:	C	D	E	F	G	A	B
voice:	none	1	2	3	4	5	6

All pitches of the same letter play the same voice.

Perc is cancelled by the next SCORE.

```
example      SCORE Perc          % select percussion mode
               D///E//D D/E///D D % two-instrument score
```



# 13 Sound instructions

This chapter explains each of the sound instructions used inside instruments: the waveforms, envelopes and number & flag instructions.

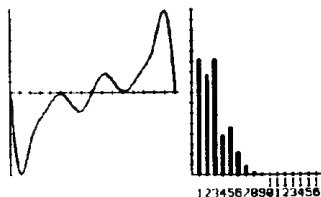
## waveforms

The following information is given for each of the waveforms:

- \* name, just as it appears on the panel without abbreviation
- \* description, necessarily vague in some cases because the effect depends on the pitch, envelope, and especially modulation
- \* list of preset instruments that use it. This is produced using the FIND command, for example, "Bright" FIND
- \* geometric plot, showing the actual shape of the waveform
- \* harmonic plot, showing levels of the first 16 harmonics. Some of the waveforms have no useful harmonic plot, so none is given

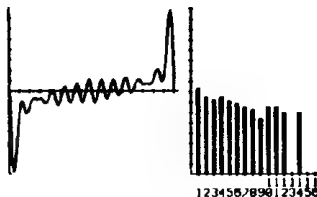
### Bright

general-purpose medium-bright  
used by: Ringsyn, Slapbass



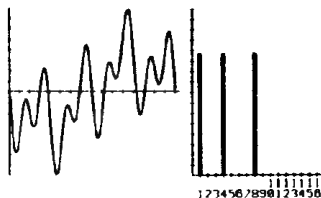
### Broad

strong and bright  
used by: none



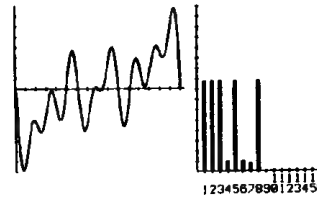
### Clear

clear, 'open', slightly glassy  
used by: Vibglock



### Hard

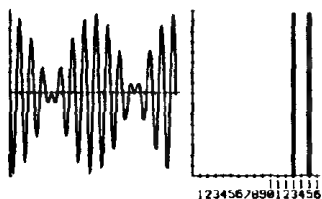
hard and bright, but rounded  
used by: Elguit



## Sound Instructions

### High

special for bell-like strikes  
used by: Vibglock



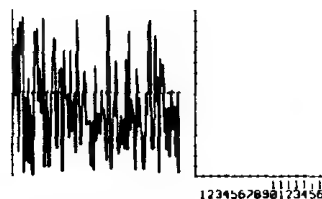
### Hollow

rounded, hollow, 'electronic'  
used by: Ironpipe, Simpleins,  
Wha



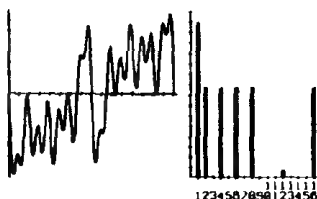
### Metal

very bright, rough metallic  
used by: Drum, Cymbal, Slapbass



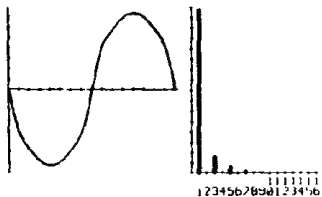
### Pipes

organ-like  
used by: Ironpipe, Organ



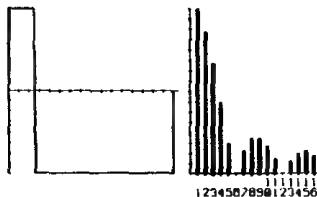
### Pure

simple, pure, flute-like  
used by: Drum, Panflute, Yakbell



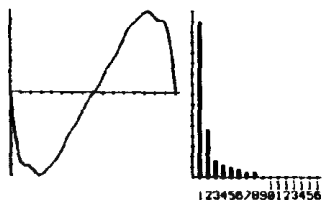
### Reedy

bright, reedy, with harsh edge  
used by: none



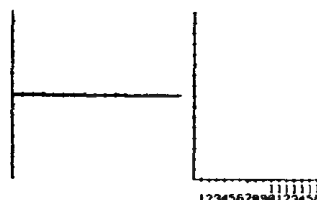
### Round

soft, rounded  
used by: Upright



### Syncer

special for strongest sync  
used by: Wha

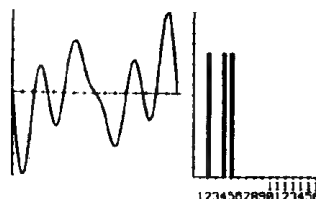


**Sharp**

warm and bright  
used by: Moog

**Watery**

rather odd, thin and 'watery'  
used by: none

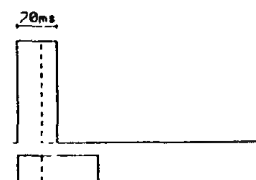
**amplitude envelopes**

The following information is given for each of the envelopes:

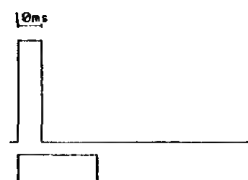
- \* name, just as it appears on the panel without abbreviation
- \* description
- \* list of preset instruments that use it. This is produced using the FIND command, for example, "Burst" FIND
- \* Simplified shape plot, with the gate period (the sounding part of the note) marked underneath. The most important section times are given in seconds or milliseconds. For many of the envelopes, the effect of a shorter gate period is shown as a dotted line.

**Burst**

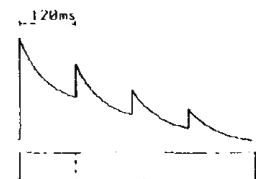
short 'burst', like a 'gated'  
studio drum sound  
used by: Drum

**Click**

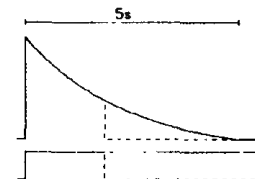
very short burst, for a single  
waveform cycle  
used by: Drum

**Echohit**

drum-like strike with three quieter  
echoes  
used by: none

**Verylong**

very long decay to silence  
used by: Ringsyn

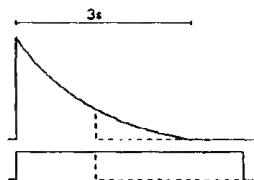


## Sound Instructions

### Long

long decay to silence

used by: Cymbal, Elguit, Vibglock



### Peaked

simple shape with a slight emphasising peak at the start

used by: Simpleins, Wha

0.25s



### Puff

soft-edged 'puff' leading into a low sustain level

used by: Panflute

0.3s

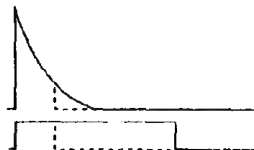


### Short

Short drum-like decay to zero

used by: Vibglock

0.5s



### Onoff

plain organ-like on-off shape

used by: Moog



### Percuss

very short, drum-like decay to silence

used by: Slapbass, Yakbell

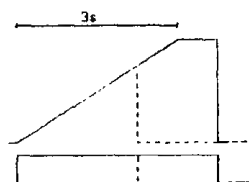
0.3s



### Reverse

reverse effect, building up and then holding at maximum

used by: none

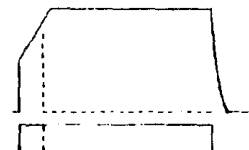


### Soft

slow blown attack and slightly slow release

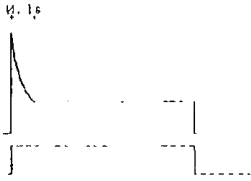
used by: Ironpipe, Organ, Wha

0.3s



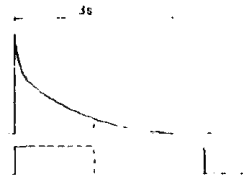
### Spike

fast, percussive decay to a quiet sustained level  
used by: none



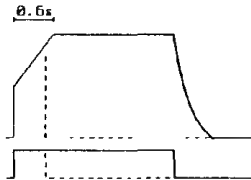
### Strike

piano-like double decay, fast then slow  
used by: Slapbass, Upright



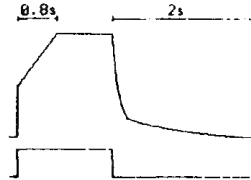
### Swell

slow string-like attack and release  
used by: Ironpipe



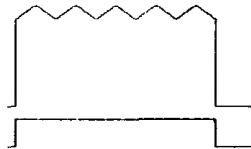
### Tailed

slow attack and two-stage reverberation-like release  
used by: none



### Tremolo

tremolo shape, with repeating loudness variation  
used by: none

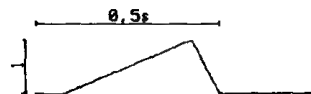


## pitch envelopes

Pitch envelopes have the same information as amplitude envelopes, but with a different shape plot. The plots all start at the gate start, so the gate period is not marked. In the case of a repeating envelope, the time marking is for a certain number of repeats. The vertical scale shows the amount of pitch change in semitone units.

### Bend

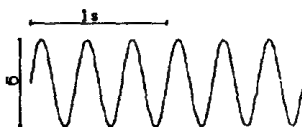
guitar-like bend, up and down at the start of the note  
used by: none



### Cycle

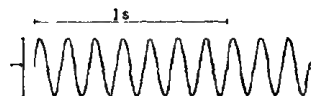
repeating up-down cycle

used by: none



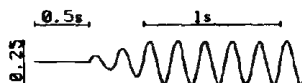
### Deepvib

deep vibrato  
used by: Wha



### Delvib

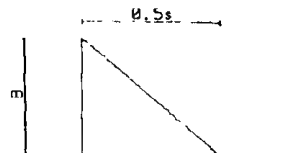
vibrato entering after the start  
used by: Elguit, Panflute, Ringsyn, Vibglock



### Drop

Fast downwards sweep

used by: none



### Flat

no pitch variation

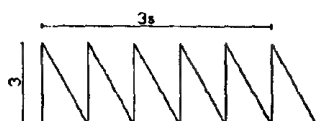
used by: various



### Pow

repeating downwards sweep

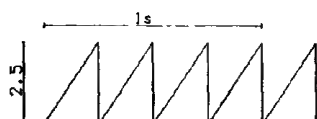
used by: none



### Ramp

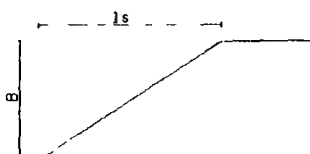
upwards klaxon-like sweep

used by: none



### Rise

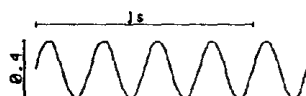
medium rise from the natural pitch to a sustained level  
used by: none



### Slowvib

slow vibrato

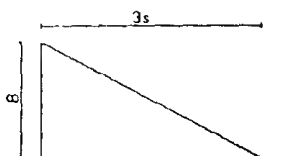
used by: Ironpipe



### Sweep

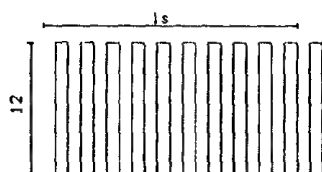
Wide slow downwards sweep

used by: none



### Trill

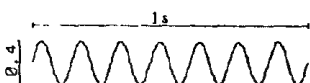
fast trilling effect to the octave above  
used by: none



### Vibrato

normal vibrato

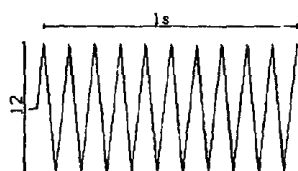
used by: Ironpipe



### Warble

intense fast warble

used by: none

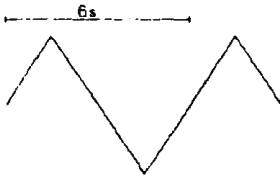


## Sound Instructions

### Wide

slow siren-like up-down cycle

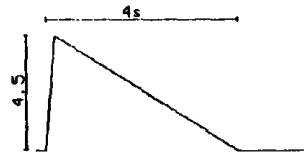
used by: none



### Wow

fast rise and slow fall back to normal

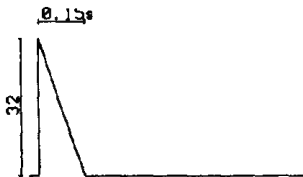
used by: Wha



### Zap

very strong and fast 'zap'  
down to the natural pitch

used by: Drum



## number and flag instructions

Of the instructions that are described here, the Notepad panel makes the CHANS and CHAN instructions fixed, but all others are variable.

instruction	function
1 CHANS	begin one-channel instrument
2 CHANS	begin two-channel instrument
ON CHANS	continue instrument
1 CHAN	select channel 1
2 CHAN	select channel 2
EVERY CHAN	select all channels
ODD CHAN	select odd-numbered channels
EVEN CHAN	select even-numbered channels
number PAIR CHAN	select numbered channel and other of pair



instruction	function	range	default
-------------	----------	-------	---------

<b>number SHIFT</b>	pitch shift	-1024 - 1023	0
---------------------	-------------	--------------	---

Set the pitch shift in 16th semitone units (192 per octave), used on one channel for setting intervals. Common values are:

192	octave up	384	two octaves up
-192	octave down	-384	two octaves down
112	fifth above	1	slightly raised

<b>number OFFSET</b>	frequency offset	-32768 - 32767	0
----------------------	------------------	----------------	---

Sets the frequency offset, used on channel 1 (usually) for enriching 'phasing' effects. A common value is 100. The unit of control is 0.0056 Hz.

<b>number AMP</b>	amplitude	0 - 128
-------------------	-----------	---------

Sets the amplitude (loudness), used on channel 2 (usually) to balance it against channel 1. The default is 128 AMP (maximum) for channel 1 and 0 AMP (off) for channel 2.

<b>number FM</b>	frequency modulation depth	0 - 255	0
------------------	----------------------------	---------	---

Sets the depth of frequency modulation, used on channel 1 (only) for complex non-harmonic (e.g. bell-like) tones. The pitch interval (SHIFT and pitch envelope) of the two channels strongly influences the sound.

<b>flag SYNC</b>	synchronisation	ON/OFF	OFF
------------------	-----------------	--------	-----

Sets synchronisation, used on channel 1 (only) to create strongly coloured (e.g. vowel-like) tones. The pitch interval (SHIFT) and pitch envelope of channel 1 controls the 'brightness' of the effect. Channel 2's waveform controls the strength, and the special waveform 'Syncer' gives the strongest effect.

<b>flag RM</b>	ring modulation	ON/OFF	OFF
----------------	-----------------	--------	-----

Sets ring modulation, used on channel 1 (only) to produce various richer, brighter and distorted sounds. The total pitch interval (SHIFT setting and pitch envelope) of the two channels strongly influences the sound.

## Sound Instructions

instruction	function	range	default
-------------	----------	-------	---------

<b>flag PHSET</b>	phase setting	ON/OFF	OFF
-------------------	---------------	--------	-----

Sets phase setting; used on both channels to give a precise start to the beginning of each note for drum and other percussive sounds.

<b>number PITCH</b>	pitch	-64 - 63	0
---------------------	-------	----------	---

Sets the pitch, used on either or both channels to set the 'natural' pitch of the instrument, that is, the pitch that will play on unpitched musical hits (the symbol X) and, if OFF PSENS is in force, all musical notes. The pitch is in semitone units (12 per octave), and middle C is 0.

<b>number PITCHF</b>	pitch (fine)	-1024 - 1023	0
----------------------	--------------	--------------	---

Sets the pitch in 16th semitone units (192 per octave), used as a fine-control alternative to PITCH. 0 is middle C.

<b>flag PSENS</b>	pitch sensitivity	ON/OFF	ON
-------------------	-------------------	--------	----

Sets pitch sensitivity; OFF PSENS is used on both channels (usually) to fix the pitch of the instrument, that is, make it unaffected by note pitches and overall tuning.

<b>flag INVERT</b>	inversion	ON/OFF	OFF
--------------------	-----------	--------	-----

Sets inversion, used on one channel to invert its signal, in order to produce an alternative phasing effect from two identical waveforms having a small OFFSET setting.

<b>number POS</b>	stereo position	-3 - 3	0
-------------------	-----------------	--------	---

Sets the stereo position, used to separate the individual channels to give a 'larger' and, in some cases, more natural sound. -3 is full left, 0 is centre and 3 is full right. To keep the overall position in the centre, the channels are put on opposite sides.

# 14 Instruments

## preset instruments

The fourteen preset instruments are listed here, each with a suggested pitch range, a short description and suggested variations. You can display the contents of any instrument with the TYPE command.

The pitch range is in AMPLE octave numbers, for example, 0: - 2: (octaves 0, 1, and 2 above middle C). Some instruments, usually drums or effects, are fixed at a particular pitch, for example: 1: .

Examples of important commands for use with instruments are:

"CHANS" FIND	% list names of all instruments
"INS1" FIND	% list users of preset instruments
"Metal" FIND	% list users of 'Metal'

### **Cymbal**        2:

Long metallic crash cymbal. The OFFSET and SHIFT control the tone.  
Hi-hat cymbal: change Long to Short

### **Drum**        -2:

Modern drum-machine-style bassdrum with strong click.  
Less click: reduce 128 AMP. Longer click: reduce -20 PITCH

### **Elguit**        -2: - 2:

Electric guitar with reverberation-like ambience.  
Thinner tone: set 0 AMP and/or 100 OFFSET

### **Ironpipe**     -2: - 2:

Iron pipe, blown across one end, with hollow resonant after-effect.  
Deeper tone: 112 SHIFT. Rougher: change Pipes to Metal, adjust SHIFT

### **Moog**        -3: - 3:

Standard synthesiser sound with rich detuned-oscillator effect.  
Faster beats: increase OFFSET. Vibrato: change Flat to Vibrato

### **Organ**        -3: - 2:

Church organ with multiple ranks of pipes.  
Softer sound: reduce 128 AMP. Hammond organ: change Flat to Vibrato

### **Panflute**     -2: - 3:

A cross between flute and panpipes.  
Less puff: change Puff to Soft. Less ambient: closer POS settings

## Instruments

### **Ringsyn**     -2: - 2:

Warm brassy synthesiser with pleasant changing tone effect.

Slower movement: reduce OFFSET. Distorted tone: various SHIFT values

### **Simpleins**     -4: - 4:

The simplest possible instrument, with a simple static tone.

### **Slapbass**     -3: - 0:

A funky, slapping bass guitar with an exaggerated clanky attack.

Less clank: reduce 128 AMP. Longer clank: change Percuss to Short

### **Upright**     -2: - 2:

Realistic upright piano, heard from nearby in a small 'dead' room.

Honky-tonk: 500 OFFSET. Backwards piano: change Strike to Reverse

### **Vibglock**     -1: - 3:

Vibraphone/glockenspiel-hybrid, with delicate strike and gentle vibrato

Softer mallets: reduce 128 AMP. Odd tone: change High to Watery

### **Wha**     -1: - 1:

Comical human voice singing 'waow'. (Set SYNC OFF to change Wow.)

Various: change Deepvib to Cycle, Wide, change Wow to Rise, Drop

### **Yakbell**     -1: - 2:

At its pre-defined pitch, a cowbell with a Tibetan flavour.

Tone: adjust 90 SHIFT. Brightness: adjust 80 FM

## type-in instruments

Here are 25 instruments for you to type in. For each one, there is:

- \* the name (to be used as the instrument word name)
- \* the suggested pitch range
- \* descriptive notes
- \* the instrument definition, as on the Notepad screen with rules

To enter an instrument, you

- \* select 'Notepad' from the Main Menu, or subsequently, enter CLEAR
- \* set the name of the instrument with NAME, for example "harps"NAME
- \* enter edit mode and type in the lines one-by-one
- \* return to command mode and enter MAKE to make the word

The spacing of items is not critical except at the start of the line, where you must enter a space if one is shown, and not if not.

You can keep instruments in text form so that they can be added to other programs. See the chapter 'Building pieces' for details.

**brassy** -2: - 2:  
mellow, thick, brass-like

-----  
2 CHANS  
2 CHAN  
200 OFFSET 128 AMP  
EVERY CHAN  
Sharp Delvib Puff  
-----

**digisyn** -2: - 2:  
Synth with hard 'clunk' strike

-----  
2 CHANS  
1 CHAN  
Short  
384 SHIFT ON RM  
2 CHAN  
Onoff  
250 OFFSET 128 AMP  
EVERY CHAN  
Bright Delvib  
ON PHSET  
-----

**echocrash** -4: - 4:  
Long crash with echoes

-----  
2 CHANS  
1 CHAN  
-600 SHIFT 3 POS ON RM  
2 CHAN  
-999 SHIFT -3 POS 128 AMP  
EVERY CHAN  
Metal Pow Long  
12 PITCH  
-----

**electom** -1: - 1:  
Electronic tom-tom, variable pitch

-----  
2 CHANS  
1 CHAN  
Metal Flat Click  
255 FM -20 PITCH OFF PSENS  
2 CHAN  
Pure Drop Short  
-112 SHIFT 128 AMP  
EVERY CHAN  
ON PHSET  
-----

**elstring** -1: - 2:  
Electric synthesised strings

-----  
2 CHANS  
1 CHAN  
1 SHIFT ON RM  
2 CHAN  
-1 SHIFT 128 AMP  
EVERY CHAN  
Hollow Slowvib Swell  
-----

**epiano** -2: - 2:  
Electric piano

-----  
2 CHANS  
1 CHAN  
Bright Flat Long  
ON RM  
2 CHAN  
Bright Flat Short  
5 OFFSET 128 AMP  
EVERY CHAN  
ON PHSET  
-----

## Instruments

**flageo** -2: - 3:  
Soft pipe up to tin whistle

---

2 CHANS  
1 CHAN  
100 OFFSET -3 POS  
2 CHAN  
128 AMP 3 POS  
EVERY CHAN  
Pure Slowvib Soft

---

**frembass** -3: - 0:  
Strong electronic FM bass

---

2 CHANS  
1 CHAN  
Hollow Flat Strike  
125 FM  
2 CHAN  
Hollow Flat Peaked  
112 SHIFT 100 OFFSET  
EVERY CHAN  
ON PHSET

---

**fuzzfade** -3: - 2:  
Fuzz, fading to flute-like tone

---

2 CHANS  
1 CHAN  
Pure Drop Peaked  
230 SHIFT ON SYNC  
2 CHAN  
Syncer Delvib Peaked  
EVERY CHAN

---

**harps** -2: - 2:  
Simple harpsichord

---

2 CHANS  
1 CHAN  
Hard Flat Strike  
ON RM  
2 CHAN  
Hard Flat Percuss  
192 SHIFT 128 AMP  
EVERY CHAN  
ON PHSET

---

**harps2** -2: - 3:  
Full harpsichord

---

4 CHANS  
1 PAIR CHAN  
Hard Flat Strike  
3 PAIR CHAN  
High Flat Percuss  
ODD CHAN  
ON RM  
EVEN CHAN  
192 SHIFT 128 AMP  
EVERY CHAN  
ON PHSET

---

**laserpan** -3: - 1:  
Long stereo laser blast

---

2 CHANS  
1 CHAN  
ON RM -600 SHIFT 3 POS  
2 CHAN  
128 AMP 0 SHIFT -3 POS  
EVERY CHAN  
Metal Pow Long

---

**metalphase** -3: - -1:

Deep, slow 'industrial' phasing

-----  
2 CHANS

1 CHAN

20 OFFSET ON INVERT

EVERY CHAN

Metal Flat Onoff  
-----**metclick** -1:

Metronome click

-----  
2 CHANS

1 CHAN

Metal Flat Click

-20 PITCH ON PHSET

EVERY CHAN

OFF PSENS  
-----**mexbass** -3: - 3:

Gentle decaying string, stereo

-----  
2 CHANS

1 CHAN

Round Delvib Verylong

1 POS

2 CHAN

Bright Delvib Verylong

100 OFFSET 128 AMP -1 POS

EVERY CHAN  
-----**noise** -4: - 4:

noise burst, with variable pitch

-----  
2 CHANS

1 CHAN

Metal Flat Burst

115 SHIFT ON RM

2 CHAN

Metal Flat Peaked

EVERY CHAN  
-----**pwbass** -3: - 1:

Strong pulse-width-modulated synth

-----  
2 CHANS

1 CHAN

Reedy Drop Peaked

60 SHIFT ON SYNC

2 CHAN

Syncer Vibrato Peaked

EVERY CHAN  
-----**robvox** -3: - 2:

Singing robot voice

-----  
2 CHANS

1 CHAN

Hollow Cycle Puff

450 SHIFT ON SYNC

EVERY CHAN  
-----**singsaw** -1: - 2:

Strong, bowed saw-like synthesiser

-----  
2 CHANS

1 CHAN

ON RM 30 OFFSET ON INVERT

2 CHAN

128 AMP

EVERY CHAN

Sharp Delvib Swell

ON PHSET  
-----**snare** -2:

'gated' studio snare drum

-----  
2 CHANS

1 CHAN

ON RM

2 CHAN

-400 SHIFT

EVERY CHAN

Metal Drop Short

-24 PITCH OFF PSENS ON PHSET  
-----

# Instruments

**swellah** -2: - 2:  
Synthesiser with swelling tone

-----  
2 CHANS  
1 CHAN  
Hollow Wide Swell  
200 SHIFT ON SYNC  
2 CHAN  
Syncer Flat Onoff  
EVERY CHAN  
-----

**woodchime** -2: - 2:  
Hard wooden chime bar

-----  
2 CHANS  
1 CHAN  
Pure Flat Long  
2 CHAN  
Hollow Flat Short  
128 AMP  
EVERY CHAN  
-----

**syndecay** -2: - 2:  
Gentle decaying timbre

-----  
2 CHANS  
1 CHAN  
Clear Sweep Onoff  
ON SYNC  
2 CHAN  
Clear Flat Onoff  
EVERY CHAN  
-----

**tarimba** -2: - 2:  
Pitched percussive with vibrato

-----  
2 CHANS  
1 CHAN  
Hard Vibrato Long  
2 CHAN  
Hard Flat Short  
192 SHIFT 128 AMP  
EVERY CHAN  
ON PHSET  
-----

**warmsus** -2: - 2:  
Warm synthesiser with strike peak

-----  
2 CHANS  
2 CHAN  
50 OFFSET 128 AMP  
EVERY CHAN  
Round Delvib Spike  
-----



## 15 Errors

When AMPLE finds a fault in a command or program, or some other problem needing your attention, it displays an **error message** and does other things like stopping the music, if appropriate. This chapter describes the effect of errors, and explains each of the messages that are likely to arise.

All error messages start with an exclamation mark to distinguish them from other messages.

When a fault is found in the input line or in an AMPLE word in the input line, a simple error message is printed, for example:

```
! Mistake
```

Any commands on the rest of the line are ignored. If a piece is playing at the time, it continues unaffected.

If the input-line error is in a user word, the message is printed with the name of the user word, for example:

```
! No number in tune
```

Again, any piece that is playing is unaffected.

If the error is in a player, that is, actually in the piece, the player number is also given, for example:

```
! Bad bar in part1 in player 1
```

In this case, the piece is stopped and all sounds are silenced.

Pressing ESCAPE is treated like an error in a player, but no player number is given.

If an error occurs during a word definition that is being entered directly at the % prompt, that is, between '[' and ']', a '!' character is printed on a line before the error, pointing to the faulty characters.

If you are in the edit mode of an editor when an error happens, the message is printed in the command area and you are left in command mode.

## Errors

If an error occurs in Notepad, either from MAKE or f1 play, then when you return to edit mode, the cursor will be positioned on the item that caused the error.

The Staff Editor works similarly, but on return to edit mode the screen will be moved to contain the cause of the error, but the cursor is not necessarily on it. The 'Bad bar' error is a special case - the cursor is always left at the end of the bad bar.

Notepad also pin-points errors that occur in existing words - if after the error happens, you GET the word named in the message, when you go to edit mode, the cursor is put at the error site.

Here are the names and explanations of all error messages you are likely to see, in alphabetical order. For advanced users, if the error comes from a module rather than the Nucleus, this is noted.

### **number for example, ! 5**

A serious fault has arisen in the system, probably as a result of memory being corrupted. If this occurs, you should restart the language from the system disc. You can save the program first, but in extreme cases this may also have been corrupted and the file will be rejected by LOAD.

### **Bad bar**

A bar line found that the total length of the preceding bar did not match the bar length set by BAR or a time signature. This usually arises from an extra or missing note in a bar, or a missing barline. If this arose from Notepad or the Staff Editor, the cursor will be put at the end of the faulty bar when you go back to edit mode.

Note that since each bar is only checked at the bar line, if you leave the very last bar line out, the system will not spot the fault.

### **Bad context**

A word has been used in an incorrect context. You probably tried to include a command-only word such as LOAD inside a word, or used a word that is only allowed inside words, like MENU, as a direct command. Types of word that can cause this error are:

word	fault
command	'command only' word inside a word definition
MENU, DISPLAY etc	'inside-word only' word entered as a command
)	no (
)K	no K(
( ... )	inside K( )K or ( )
K( ... )K	inside K( )K or ( )
READY	in a player (can only be used at keyboard)

**Bad element**

An attempt was made to access an array element outside the range dimensioned. You will not normally create arrays of your own, so this error should not arise.

**Bad hex**

The first character after & was not a valid hex digit.

**Bad mode**

There was not enough free memory for the mode requested by MODE or a module. This usually arises from the Staff Editor when the program is too big to allow the Editor's graphics screen to be used.

If the error comes from other than the Staff Editor, you can enter the command COMPACT to make sure all the memory is available, and try again (STAFF, LOAD and SAVE do COMPACT automatically).

On BBC Microcomputers with shadow RAM, this error does not appear.

**Bad name**

There was a fault in the name given for a new word, either in '[', NAME or RENAME. It was probably too long (longer than 15 characters) or null ("").

**Bad player number**

An invalid player number was given. The range for SHARE is 0 to 10.

**Bad program**

The LOAded file was not an AMPLE Nucleus program produced by SAVE. It is possible to get the 'Too big' for the same reason. If either of these or any other error arises while loading a program, you are left with a clear program, as after NEW.

**Bad ROM**

The Nucleus ROM image is faulty, due to a fault in the computer or the ROM IC itself. Before suspecting the ROM, you should try it on another computer. The ROM image is checked for faults on \*AMPLE (start-up from the system disc), AMPLE and BREAK inside AMPLE.

**Bad string**

There was no closing double-quote character in the string.

**Bad structure**

A structure such as FOR( ... )FOR was incomplete, for example, the )FOR was missing.

**Division by zero**

An attempt was made to divide by zero. Probably, you used +L, -L or ' when the length setting was 0, or inside a chord.

### **Escape**

The ESCAPE key was pressed. ESCAPE stops all players and sounds.

### **Extra number**

There was a number left over after the command had been carried out.

This arises when there is a surplus number on the line or in a word on the line, that is, a number which is not used by following words. For example,

```
3 3 VOICES          (two numbers instead of one)
24,bDc 12DE 24,G    (missing comma after 12)
```

### **Extra string**

There was a string left over after the command had been carried out.

This usually means that you gave a string where one was not required, for example:

```
"tune" MAKE
```

### **In use**

The word you tried to delete with DELETE was in use by another word. You can find all uses of the word using the FIND command.

### **Mistake**

Some characters on the input line were not understood, i.e. were not recognised as a word, number or string.

If you mistype a command, the system will interpret as much of it as it can in terms of the existing user and system words before it is forced to give the 'Mistake' message.

This can sometimes arise through failure to separate two words by a space, for example, 'VOICESHOW' would be interpreted as 'VOICES HOW', and the error would be given on reaching the 'HOW'.

### **No number**

A number was missing. This is usually the result of leaving an argument out, for example

```
VOICES              (should be, for example, 1 VOICES)
```

### **No room**

There was not enough free memory for the operation. This usually means that all available memory is in use, and the program cannot be made any bigger. The system needs working memory for many operations so when you are near the limit, this can occur at any time. Remember

that the following items use up memory:

- \* the program
- \* the editor selected from the menu
- \* the graphics screen of the Staff Editor (on non-shadow computers)
- \* each player in the piece, while it is playing only
- \* the computer's operating system, including any additional ROMs you have fitted to your computer

When a piece ends, its players are left in existence, using up memory. You can free this by pressing ESCAPE, entering STOP, or stopping the program from the menu.

Some remedies are as follows:

- \* reduce the program size
- \* if in the Staff Editor, switch to Notepad (non-shadow computers)
- \* remove or disable ROMs that use memory at all times

If you get the error during editing, it can be worth entering the command COMPACT to get the last drop of available memory.

When this error comes from an attempt to add notes to the Staff Editor staff, it means that the Editor's own fixed limit (about 250 notes) has been reached.

### **No string**

A string was missing. You have probably left out an argument, or put it in the wrong place, for example

```
SAVE "temp"
```

### **No such item**

There is no word or module of the given name. This can arise wherever you specify the module or word name in a string, for example "tune"GET.

Note that when specifying a word or module, you must enter the name with the correct upper- or lower-case of letters.

### **Too big**

There is not enough free memory to load the program or module. If the loaded file was not a program, you may get this error rather than the expected 'Bad program'.

If the amount of memory available for programs and modules changes for some reason, you may end up with too little for a program or module that previously loaded successfully. See the error message 'No room' for more information.

### **Too many channels**

All sixteen channels were in use when a request was made for some channels, probably when putting an instrument on a voice. You will only get this error when using instruments with more than 2 channels each, because you will otherwise first reach the maximum number of voices (eight) and get the 'Too many voices' error. See 'Too many voices'.

This error is issued by the M5 module.

### **Too many characters**

The maximum total length of strings was exceeded. You probably have a very long string in your program, and started it with a very long line of commands.

### **Too many levels**

This is only likely to arise if you have mistakenly included the name of a word inside itself, for example:

```
"part1" [ ... part1 ... ]
```

### **Too many numbers**

This happens if you have a large quantity of extra numbers inside a program: in practice it is only likely if you have an extra number inside a loop, so that it accumulates each time around.

### **Too many strings**

The capacity of the string stack was exceeded. It can hold 16 strings. You are not likely to get this error.

### **Too many voices**

All eight voices were already in use when an attempt was made to get another. This error happens when you try to put instruments on more than eight voices. You may get this unexpectedly if you try to set up a voice with instrument at the keyboard when a piece has finished but its players and voices are still in existence. To free the voices, stop the program by pressing ESCAPE, entering STOP or selecting 'stop program' from the menu.

### **Too many words**

The maximum number of user words allowed had already been reached - the maximum is 125.

### **Duplicate name**

(This message is not an error but a warning (it appears without a !), but is included here for completeness). The new name specified to RENAME was already in use for another user word, but the operation is still carried out. All subsequent references to the name will refer to the new word. You can use RENAME again to change the name.

## 16 Editor controls

This chapter gives you a summary of special key controls in the three editors. A key strip is included with the Installation Guide.

### Notepad

#### general

left, right, up, down	move cursor
f1 play	play music or instrument
f2 text/panel	switch between text and panel modes

#### text mode (line cursor)

DELETE	delete character before cursor, moving back
COPY	start copying
RETURN	move to start of next line/end copying
f6 insert line	insert line before cursor, moving lines down
f7 delete line	delete current line, moving lines up
f8 insert char	insert space before cursor, moving part line right
f9 delete char	delete character at cursor, closing up

#### panel mode (block cursor)

SHIFT up/down	increase/decrease number by one / turn flag ON/OFF
SHIFT left/right	increase/decrease number by ten
SHIFT (on word)	select waveform or envelope from menu
£ (SHIFT _)	set number to zero

### Staff Editor

#### general

DELETE	delete character before cursor
COPY	delete character at cursor
SHIFT COPY	delete to bar line or end
left, right	move by one character
f1 play	play music
f2 copy	copy end of music
ESCAPE	stop music
up	move 'up' to text mode (shown by line cursor)
down	move 'down' to symbol mode (shown by block cursor)
CTRL left/right	move by one screenful
CTRL up/down	move to start/end

**symbol mode** (block cursor)

SHIFT up/down      raise/lower pitch of note  
SHIFT left/right    shorten/lengthen note or rest

@	insert chord note	space	insert space
c	insert clef	e	insert end line
^	insert rest	+	insert sharp
-	insert flat	=	insert natural
k	make key signature	~	insert tie/slur
.	insert dot	3	make triplet
2	insert duplet figure		insert bar line
d	insert 6/8 time signature		
t	insert 3/4 time signature		
q	insert 4/4 time signature		

## Mixing Desk

**general**

left, right, up, down	move cursor
r	run piece
SPACE	on/off pause
g	on/off group
f	fast wind
/	step on by one beat when paused
RETURN	play hit on current voice(s)
numbers 1-8	play hit on numbered voice
b	set/clear brackets of current voice(s)/tempo/tune
i	set/clear instrument brackets
p	set/clear pan brackets
v	set/clear volume brackets

**cursor on tempo/tune**

SHIFT up/down      increase/decrease by one  
SHIFT left/right    increase/decrease by ten  
COPY                reset to normal

**cursor on instrument/voice controls**

SHIFT                select instrument (instrument)  
SHIFT up/down      increase/decrease volume (controls)  
SHIFT left/right    move left/right (controls)



# 17 Glossary

This chapter presents a useful glossary of musical, sound and AMPLE terms.

**accent** any variation that makes a note stand out from others, that is, emphasises it.

**accidental** any sharp, flat, or natural before a note, that is not in the key signature.

**ADSR** a simple form of envelope usually used for amplitude. It stands for attack, decay, sustain, release - the four basic parameters needed to model the amplitude envelopes of the most natural instruments.

**amplitude** a measure of the loudness of a signal at a particular instant.

**arpeggiation** playing of the notes of a chord separately, in sequence.

**aperiodic (adjective)**  
see Non-periodic.

**articulation** the subject of how notes are pronounced.

**attack** the build-up of amplitude from zero at the start of a note. The sound of a piano has an immediate attack, whereas that of a flute or organ is longer.

**bar** a division of musical time. Every bar has the same total of note lengths, and therefore, with constant tempo, each lasts for the same amount of time.

**bar line** the instruction that ends one bar and starts the next.

**beat** a regular pulse you can sense even when the notes are shorter or longer than it.

**channel** the basic sound-generating part of the Synthesiser. There are sixteen channels and each produces a single sound signal with its own pitch, tone, loudness and stereo position.

**chord** a group of notes that sound at the same time, usually starting and ending together. The notes are written together as a group but they play on different voices.

**chorusing** an effect used to make sound richer by playing similar waveforms together with slightly different and often varying frequencies.

**clef** sign to mark the axis of the graph indicating the pitch of a particular position, and the others follow from it.

**conductor** directs the players so that they play their parts in time with each other.

**cycle** one complete repetition of a pattern, such as a wave form.

**decay** in particular, the decrease in amplitude after the start of a note. The sound of a piano has a pronounced decay, whereas that of an organ has none.

**dot** a staff instruction that increases the length of a note by half. Two dots add three-quarters to the length.

**duplet** a group of two notes played in the time of three.

**duration** the amount of time between two events (in particular between one group of sound-changing commands and the next) or, as a result of this, the amount of time that an effect lasts for.

**dynamic level** elements of sound (usually amplitude and timbre) that change over the course of time. How loud or soft a passage is to be played.

**echo** a delayed repetition of a sound. One or more echoes may be used to enrich the original or for musical effect.

**envelope** the shape of some aspect of a sound as it varies over time. The commonest types are the amplitude envelope and the pitch envelope.

**flat** an instruction that lowers the pitch of the next note by one semitone, or if in the key signature, does the same for all notes of a particular name. Also name of a pitch envelope which produces a non-varying pitch.

**frequency** a scientific measure of how 'high' or 'low' a sound is: the rate of repetition of the sound's vibration pattern. Frequency is stated in Hertz(Hz); the number of cycles per second. Middle C has a frequency of 261.6 Hz.

**frequency modulation (FM)** rapid variation of the frequency of one signal by another signal, producing new timbres. FM sounds are very complex (like bell and gong sounds) and often have ambiguous pitches.

**fundamental** the lowest sinewave component of a periodic waveform (the first harmonic). It sounds at the same pitch as the complete waveform, and is often the strongest harmonic (that is, has the greatest amplitude).

**gate** the signal that controls an envelope generator. Gate 'on' sends the envelope to the start of its 'on' section. In the case of most amplitude envelopes, the 'on' section lets the sound through and 'off' section turns it off. Notes send gate 'on' signals, rests send gate 'off' signals, and ties send no gate signals.

**glissando** A rapid slide in discrete steps, up or down, through consecutive pitches.

**geometric plot** diagram of the shape of a waveform.

**harmonic** a single sinewave component of a periodic waveform. Any periodic waveform can be thought of as a series of sinewaves of different frequencies added together. These are its harmonics, and their relative amplitudes determine the timbre of the sound.

**harmonic plot** diagram showing the strengths of the harmonics in the waveform.

**Hertz** the unit of frequency. One hertz represents one cycle per second. The unit is abbreviated to Hz.

**hit** in a score, the pitchless equivalent to a note, such as plays on drums.

**instrument** the object that determines the sound used by a musical voice. The instrument is responsible for aspects of the sound that are fixed from note-to-note, such as timbre, but not those that vary in the playing of the music, such as pitch.

**interval** the pitch difference between two notes/pitches.

**key** the particular set of pitches that a piece uses for the notes of the scale, described as the starting note of the scale (the 'key note') and the type of the scale, major or minor.

**key signature** instruction to modify the pitches of notes to suit the key of the music following. The normal pitches of the notes suits the key of C major, but since the interval between adjacent notes varies up the scale, the note pitches need to be modified for every other key to keep the same relative pitches at the new starting note.

**ledger line** lines added above and below the staff for pitches outside the range of the staff.

**legato** a style of playing where successive notes are smoothly connected, with no gap between them.

**modulation** rapid variation of a signal's parameter by another signal, producing new timbres. In music, a modulation is a change of key in the middle of a piece of music.

**natural sign** an staff instruction that returns the note to its original pitch.

**noise** a particular type of sound that has no identifiable pitch. The sounds of waterfalls, waves, hissing steam and cymbals are all noise-type sounds. Noise can be thought of as a mixture of an infinite number of different frequencies, with their relative amplitudes determining whether the noise is 'high' or 'low', rough or smooth etc. In practice, noise is produced by ring modulation using pseudo-random waveforms.

**non-periodic (adjective)** having no identifiable repeating period. Non-periodic waveforms are typical of complex vibrating objects like bells and gongs.

**note** a sound of a particular pitch and length; the fundamental unit of most music. The note names A to G refer to seven particular semitone pitches in any octave.

**note style** the details of how notes are to be played, including the gap between notes.

**octave** a group of seven adjacent letter-name pitches, for example, A to G, or the interval between two adjacent pitches of the same name.

**ornament** decoration to a note which adds interest without changing the basic musical effect of the note.

**overtone** another word for harmonic. It can also mean those partials that are above the perceived pitch of a non-periodic tone.

**panning** the panoramic movement of sound, that is, from side-to-side in the stereo field.

**part** a 'horizontal' sub-division of a score. All the parts play together at the same time to make up the complete piece. Parts are often named after their musical ranges (soprano, tenor), musical functions (rhythm, tune), or the instruments that play them (guitar, vocal).

**partial** a single sinewave component of a waveform. If the frequencies of the components are integral multiples of a fundamental frequency, the waveform is periodic and the components are usually called harmonics. If no audible fundamental frequency can be identified, the waveform is non-periodic and the components are usually called partials.

**peak** the greatest amplitude of a sound wave or signal.

**periodic (adjective)** repeats exactly after an identifiable interval of time. Periodic waveforms are produced by simple vibrating objects such as the strings and air columns of musical instruments.

**phase** a position in the cycle of a waveform, stated in degrees (a complete cycle is 360 degrees). A 'phase difference' is the separation between related points in the waveforms of two signals playing together. Setting the phase of a waveform is starting it off from a particular point in its cycle.

**phasing** the sound produced by two similar waveforms with very slightly different frequencies. As the phase difference slowly changes, different harmonics cancel out, giving a jet-plane-type swooshing sound.

**phrasing** the grouping of notes into expressive divisions of music.

**pitch** the musical measure of how 'high' or 'low' a sound is. The pitch of a note is usually described as octave, measured from the octave range above middle C, and note name (C, D# etc), indicating the semitone within the octave.

**player** an object that plays a single scored part of music (or sequence of instructions), alongside other players. A 'master' player always exists, to handle instructions typed at the keyboard.

**portamento** continuous pitch slide usually between two notes.

**release** that part of the sound of a note which appears after the note is released. Most natural instruments have an immediate release, but in the case of percussive instruments, the whole sound can be considered to be the release.

**resonance** the boosting of certain frequencies in a sound, changing the tone.

**rest** a period of silence in a musical voice. Rests have lengths just like notes, and it is useful to think of them as notes which are 'off' and therefore have no pitch. Some instruments have sounds that

## Glossary

carry on after the note has officially finished (see release), so there may not in fact be silence while a rest plays.

**reverberation** the effect of a large enclosed space, such as a concert hall, on the character of a sound.

**ring modulation (RM)** a rapid variation of the amplitude of one signal by another, named after the ring-shaped electronic circuit first used to produce it. The new timbres produced by the form of ring modulation used here include complex distorted timbres and pitchless noise-based sounds.

**scale** a series of pitches strung together to form a musical ladder structure. The group of different note pitches in one octave, often played up or down in order. The commonest types of scale are major and minor, which each have seven notes to the octave.

**score** a full set of instructions for playing a piece - the musical 'program'.

**semitone** a small unit of pitch; the smallest used in most music. The smallest difference between adjacent notes in a scale is one semitone, for example, between E and F. Two semitones equal one tone.

**sharp** a staff instruction that raises the pitch of the next note by one semitone, or if in the key signature, does the same for all notes of a particular name.

**sinewave** the simplest waveform. The sound of a sinewave is very pure and plain, with no brightness, colour or distortion. The on-the-hour pips of the Greenwich time signal and the TV end-of-transmission tone are both sinewaves.

**slur** two notes played in a connected, smooth fashion, without any gap at all.

**staccato** the written length of a staccato note is roughly half sounding, followed by half silence.

**staff** on which pitches are written. A double staff has ten lines covering a wider range. The plural is 'staves'.

**sustain** the holding-on of the sound while the note plays. It can also mean a continuing sound after the note has finished (release) as in the case of sustain stops on an electronic organ. In an ADSR envelope, the sustain level is the level to which the amplitude eventually settles when the note is held on.

**synchronisation ('sync')** a form of modulation where the waveform of one signal is distorted by synchronising it to another, producing strongly-coloured (for example vowel-like) timbres.

**tempo** the speed of a piece of music. In a computer system the tempo depends on timebase period, which can be varied after the piece has been scored, even while it is playing.

**tie** an instruction that joins a note to the previous one, making one long note.

**timbre** the 'tone' or 'quality' of a sound, as opposed to its pitch, loudness, envelopes etc. Waveform and modulation determine the timbre of a synthesised sound.

**timebase** the internal reference which controls all durations in a piece of music, like a conductor's baton.

**time signature** an indication of the length of the bar for the music following.

**transposition** shifting the pitch of music up or down, so it plays in another key.

**tremolo** a repeating variation in amplitude of a note as a feature of the instrument. It is characteristic of the sound of the flute. In natural instruments it is often accompanied by vibrato, and the two effects are often confused.

**triplet** a group of three notes played in the time of two of the same written length.

**vibrato** a repeating wavering imposed on the basic pitch of each note by the instrument. The rate is usually between two and six cycles each second.

**voice** an individual music-playing unit that plays a single note at a time. To play chords, you need more than one voice. Each voice can have its own instrument and its own notes to play. A 'musical voice' refers to a line of notes in the score that plays on a single voice.

**waveform** the shape of a sound's vibration pattern. The waveform determines the timbre of a sound. A periodic sound has an easily-identifiable waveform since it repeats on each cycle, but a non-periodic sound has no identifiable repeating cycle and therefore no fixed waveform.





# Index

accent - '	101	delay, negative	104
accidentals	39,101	delete	
ADFS	89	bar	30
adjusting whilst playing	72	editor data - CLEAR	93
amplitude	64	program - NEW	21,96
arpeggiation	106	user word - DELETE	94
articulation effects	106	DFS	89
		discordant intervals	68
backwards-playing notes	106	dotted notes	31
bar line	26	double staff	22
boot file - !BOOT	88	drive, changing	88
brackets	114	duplets	31,106
		dynamic level, range	39,102,
case, upper and lower	59		117
catalogue user words		echo	119
- SHOW	99	editing	
CHAN	70,75	text	41-44,
CHANS	74		94,149,
chord			150
broken	103	with the Staff Editor	21-40
building	37	entry, tune	22
of rests	104	envelopes	60,113
clearing text - CLEAR	23,42,	EVEN	75
	65,66;	executing files - *EXEC	89
	93	exiting current editor	
clef	23,108	- QUIT	97
comments	93	flat	28,106
compilation - MAKE	95	FM	70
conductor part	121	gate period	32
copying groups of notes	29	grace notes	106
copy files in AMPLE		group mode	73
- *COPY	90	hits	104
		holds	103

inserting lines and chars	42	naming a tune - NAME	24,96
instrument		natural	28,109
continuation	74	negative	
examination	57	delay	104
many-channel	75	length	106
new	60,65	name	24,43, 61
playing	58	natural	25
key signatures	28,116	noise	69
		note	
leading chords	104	additional	102
leaving editor - QUIT	97	bracketed	102
legato	32	dotted	31
length		entry	22
gate	33	main	102
suggested values	105	style	32
listing		Notepad	41,57
user program - WRITE	100	notes A-G, a-g	114,115
user words - SHOW	99		
word contents - TYPE	99	octave	
loading		jumps	101
editor data - GET	94	setting	59
program - LOAD	25,95	ODD	75
		OFFSET	64,69
memory		OSHW	89
compaction - COMPACT	94		
usage	89	PAIR	75
usage report - MEM	95	pan control	72
MENUDISP	87	panel	
menus, use of	87,95, 96	design	73
minus sign	107	mode	57
Mixing Desk	38,72, 103	modification	73
		percussion	37
modifiers	35	position, stereo - POS	72
modulation	67	pitch	22
moving about	29	PITCH	67,123
MPREFIX	88	phase setting - PHSET	65
musical paintbrush	22	playing	
		chords	36
		instruments	58
		tunes	24
		pling - !	101
		PREPARE	89
		preset words and	
		instruments	61
		printing it out	100

RAM	89,96	waveforms, trying out	60
readying system - READY	21,120	words, user	
rests	25,59,	catalogue - SHOW	99
	103	creating	24,43,
dotted	31		60
on chords	36	compiling - MAKE	25,43,
restart/ready system	9,93		95
riff	110	deleting - DELETE	94
ROM	89,97	listing - TYPE	99
		modifying	44,61
searching for words		naming	43
- FIND	24,94		
sensitivity, pitch		X	104,122
- PSENS	66		
shadow RAM	89		
sharing other player's			
voices	72		
sharp	28,105		
SHIFT	64,69		
silencing the system	69		
SIMPLEACT	120		
spaces	29		
spooling files - *SPOOL	89		
staff, double	22		
Staff Editor	21-40		
stereo position - POS	72		
structure	30		
style, note	32		
synchronisation - SYNC	70		
tempo	24		
text entry	42		
(see 'editing')			
ties	30,		
	120-122		
time signatures	26		
'To the Manor Born'	73		
tone, change of	68		
transposition, set - @	110		
triplets	31,106		
trying things out	60,61		
tune entry	22		
voices - VOICE and VOICES	24,36,		
	58,62,72		





